UNIVERSITY OF CAPE TOWN

ANALYSIS OF A DEEP NEURAL NETWORK FOR MISSING TRANSVERSE MOMENTUM RECONSTRUCTION IN ATLAS

Author:
Matthew LEIGH

Supervisor:
Dr. Sahal YACOOB

Co-Supervisor:
Dr. Christopher Young

A thesis submitted in fulfilment of the requirements for the degree of Master in Science

in the

Department of Physics

June 30, 2020

Declaration of Authorship

I, Matthew LEIGH, declare that this thesis titled, "ANALYSIS OF A DEEP NEURAL NETWORK FOR MISSING TRANSVERSE MOMENTUM RECONSTRUCTION IN ATLAS" and the work presented in it are my own. I confirm that:

- This work was done wholly while in candidature for a research degree at this University of Cape Town.
- Where I have consulted the published work of others, this is always clearly attributed and the source is always given.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed.

Signed: Signed by candidate

Date:

"Its you! Despite everything. Its still you."

Toby Fox

Abstract

Matthew LEIGH

ANALYSIS OF A DEEP NEURAL NETWORK FOR MISSING
TRANSVERSE MOMENTUM RECONSTRUCTION IN ATLAS

The ATLAS detector is a multipurpose particle detector built to record almost all possible decay products of the high energy proton-proton collisions provided by the Large Hadron Collider. The presence and combined kinematics of unobserved particles can be inferred by the observed momentum imbalance in the transverse plane. In this work, a deep neural network was trained using supervised learning to measure this imbalance. The performance of this network was evaluated in MC simulation and in $43\,\mathrm{fb}^{-1}$ of data recorded at ATLAS. The network offered superior resolution and significantly better pileup resistance than all other pre-existing algorithms in every tested topology. The network also provided the best discriminator between events that did and did not contain neutrinos. The potential gain in sensitivity to new physics was demonstrated by using this network in a search for the electroweak production of supersymmetric particles. The expected sensitivity to observe the production of said particles was increased by up to 26%.

Acknowledgements

I would like to thank the following individuals and groups, without whom the completion of this dissertation would have been impossible.

Firstly, I would like to thank my supervisors Dr. Sahal Yacoob and Dr. Christopher Young. Sahal, thank you for giving me the incredible opportunity to join the world of HEP and ATLAS. Thank you for your expertise, your patience, and your overall support over the past two years. I also appreciate our more informal chats about physics and my well-being. Chris, thank you for the help you provided especially during my trips to CERN, and for swooping in to save the day whenever I ran into a technical problem that threatened to derail the project. And to both of you, thank you for encouraging me even when deadlines were missed, and when chapters were handed in with more words than were needed for the entire project.

I would like to also extend my gratitude to the University of Cape Town Postgraduate Funding Office for funding my studies, to the SA-CERN collaboration for funding the trips to CERN, and to the University of Cape Town High Performance Cluster team, who provided the facilities needed to train the thousands of neural networks used in this project.

I would also like to thank my friends and officemates, Kevin Barends, Ryan Atkin, and Chilufya Mwewa. On top of being hugely supportive to me and my work, you would drop what you were doing whenever I needed help understanding something about ATLAS. I have learned more from you three than from any textbook or lecture.

To my friends who gave up their free time to help find any one of the thousands of mistakes in my first draft. This text would be unreadable without you. Thank you, Charlie Schleich, Dylan Jones, Albie Du Toit, Ben Warren, Alasdair Falconer, Tomas Bruce-Chwatt, Mayhew Steyn and Thejal Mathura. I believe I promised each of you a drink for every mistake you found. You all found so much that I look forward to paying you back over the rest of my life.

Finally, I would like to thank my parents. On top of the long sessions editing this work and patiently sitting while I talked physics excitedly at you, you believed in me and always cheered for my success.

Contents

Chapter 1

Introduction

The Large Hadron Collider (LHC), built under the France-Switzerland border, produces proton-proton (pp) collisions at a centre-of-mass energy of up to 13 GeV. The products of these collisions are captured by the ATLAS detector, which can observe nearly all the outgoing stable or long-lived particles. There are however a couple of notable exceptions. Neutrinos, belonging to the Standard Model of Particle Physics (SM), pass through normal matter unimpeded and therefore travel straight through the ATLAS detector without leaving any directly measurable signals. The presence of these undetected particles can be inferred from the outgoing momentum imbalance in the plane perpendicular to the original proton beams.

Linear momentum must be conserved during the collisions. Furthermore, the total momentum perpendicular to the beamline, defined as the transverse momentum, of the colliding protons is essentially zero. The combination of these two statements imply that the net transverse momentum of all products of the collision must also sum to zero. Any deviation from zero implies that an undetected particle carried momentum away from the interaction. The negative vectorial sum of all observed momenta therefore serves as an experimental proxy for the total transverse momentum of undetected particles. This value is referred to as the missing transverse momentum or $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$.

Measuring $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ or its magnitude $E_{\mathrm{T}}^{\mathrm{miss}}$ with a high degree of accuracy is critical for the understanding of many physical processes which take place at the LHC, and thus the variable is an important feature in many ongoing analyses. It is particularly important for processes which involve the weak force [1–3], due to the production of neutrinos. Historically $E_{\mathrm{T}}^{\mathrm{miss}}$ played an important role in the discovery of the Higgs boson by ATLAS in 2012 in the decay channels $H \to WW$ and $H \to \tau\tau$ [4]. The $E_{\mathrm{T}}^{\mathrm{miss}}$ is used directly for most measurements which involve the W bosons [5, 6] or top quarks [7]. Even for processes that do not involve the production of neutrinos, $E_{\mathrm{T}}^{\mathrm{miss}}$ is often used for event selection to increase signal purity.

Beyond the Standard Model (BSM) there exist many theorised particles which, like neutrinos, would not leave signals in the detector. Therefore, $E_{\mathrm{T}}^{\mathrm{miss}}$ measurements

are crucial for testing the validity of these models as well. These theoretical particles include some weakly interacting supersymmetric (SUSY) particles and particles belonging to models attempting to account for dark matter.

The reconstruction of $E_{\mathrm{T}}^{\mathrm{miss}}$ is very challenging since it involves a complex combination of hundreds of thousands of channels from every sub-detector in ATLAS. Any mismeasurement of just one of the visible objects would result in a perceived imbalance, which was not created by the production of undetectable particles. This is known as fake $E_{\mathrm{T}}^{\mathrm{miss}}$. The reconstruction is therefore very sensitive to the misidentification of particles, errors in reconstructed particle tracks, and uncertainties in the calorimeter readings. The $E_{\mathrm{T}}^{\mathrm{miss}}$ measurement is further degraded by the erroneous inclusion of signals from additional pp interactions which occur in the same, subsequent or previous bunch-crossings (pileup). This effect is only expected to get worse as the luminosity of the LHC increases over the next few years, and significantly so after the major upgrade of the LHC due in 2025 which will increase the instantaneous luminosities by a full order of magnitude [8].

Over the past several years, the ATLAS experiment has employed several algorithms to reconstruct $E_{\mathrm{T}}^{\mathrm{miss}}$ [9–11]. These algorithms differ in the information used to create the negative vectorial sum, and offer varying levels of signal efficiency and pileup suppression. Due to the vastly varying final states produced in ATLAS, the performance of these algorithms is topology dependent and the algorithm that offers the best resolution in one environment may not be the ideal choice for another.

The aim of this project is to investigate the use of machine learning, particularly in the form of deep artificial neural networks (ANN), to combine the various $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms to form a single, most accurate measurement of $E_{\mathrm{T}}^{\mathrm{miss}}$ for all event topologies. ANNs have proved to be incredibly useful tools in reproducing highly complex nonlinear processes and have thus been applied in numerous fields such as facial recognition [12], data mining [13], medical diagnosis [14], and object recognition [15]. The successful application of machine learning has already been observed in many aspects of high energy physics, especially in reconstruction [16].

This thesis has three major components.

First, it studies how an ANN can be trained for $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Exploring what information must be provided to the neural network for it to produce an accurate working model that best suits the needs of the ATLAS collaboration.

Secondly, it provides an in-depth look at the performance of a final model in both data and Monte Carlo (MC) simulated samples. This investigation uses the standardised methods used by the ATLAS collaboration when evaluating any $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm. It also looks at how the performance of the ANN is affected by its learning process and the data used to train it.

Finally, the trained neural network and the default ATLAS $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction algorithm are compared when applied to the search for evidence of particles belonging to BSM physics. This study attempts to demonstrate the potential performance gain achievable by the neural network in a typical analysis.

Chapters of this dissertation are structured in the following manner. Chapter 2 discusses the foundational and theoretical background of SM physics and its SUSY extension. Chapter 3 reviews the theoretical concepts of deep learning and introduces the various techniques and optimisations used in this project. A summary of the LHC and concepts pertaining to pp collider physics is presented in Chapter 4. Since $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction requires the consideration of all observed signals, the current ATLAS detector and all its components are discussed in Chapter 5. The standard procedure used to convert raw detector signals into calibrated physics objects is summarised in Chapter 6. An overview of the various $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction methods used at ATLAS as well as the standard methods used to determine their performance is shown in Chapter 7. Chapter 8 details all data and MC samples used in this dissertation. The training procedure and philosophy used to develop a final working ANN is covered in Chapter 9 and the evaluation of that model is then conducted in Chapter 10. The results and discussions around the application of the ANN in a typical search for SUSY signals are presented in Chapter 11. Finally, the last chapter provides a summary of the investigations and results obtained in this thesis and presents possible avenues for future work.

Additional plots and supporting material can be found in Appendix A.

Chapter 2

Theoretical Background

This chapter serves as an outline of the physical theories required to understand the nature of the work presented in the following chapters. However, since the focus of this project is on information processing and machine learning, only a brief overview of the Standard Model and one of its extensions is required. Unless otherwise stated, References [17–20] were used to write this chapter.

2.1 The Standard Model of Particle Physics

The Standard Model of Particle Physics (SM) is one of the most successful and widely accepted theories in physics [17]. Developed in many stages throughout the latter half of the 20^{th} century, it aims to describe the nature of all elementary particles in the universe and all interactions that can take place between them. In the SM, all matter is comprised of point-like particles called fermions. These fundamental particles have no internal structure and are only able to interact via the exchange of additional force carrying particles called vector bosons. Each vector boson corresponds to one of the three fundamental forces described in the SM; the strong force, the weak force and the electromagnetic force.

All particles are categorised according to their spin. All fermions have half-integer spin and can be further classified as quarks or leptons. As shown by the columns on Figure 2.1, there are three generations of fermions each containing two quarks, a charged lepton, and a neutral lepton. Each following generation contains particles which are more massive and less stable, but otherwise identical to the ones in the previous generation. The exceptions to this rule are the neutral leptons, also called neutrinos, which do not decay. The SM predicts neutrinos to be massless, but this was disproven via the observation of neutrino oscillations in 1998 [21]. Stable matter encountered in everyday life is made up of the fermions in the first generation: the up quark, down quark (which together create protons and neutrons), and the electron.

The SM is a renormalisable quantum field theory (QFT) [23] developed to adhere to the principles of quantum mechanics and special relativity. Being a form of QFT, the

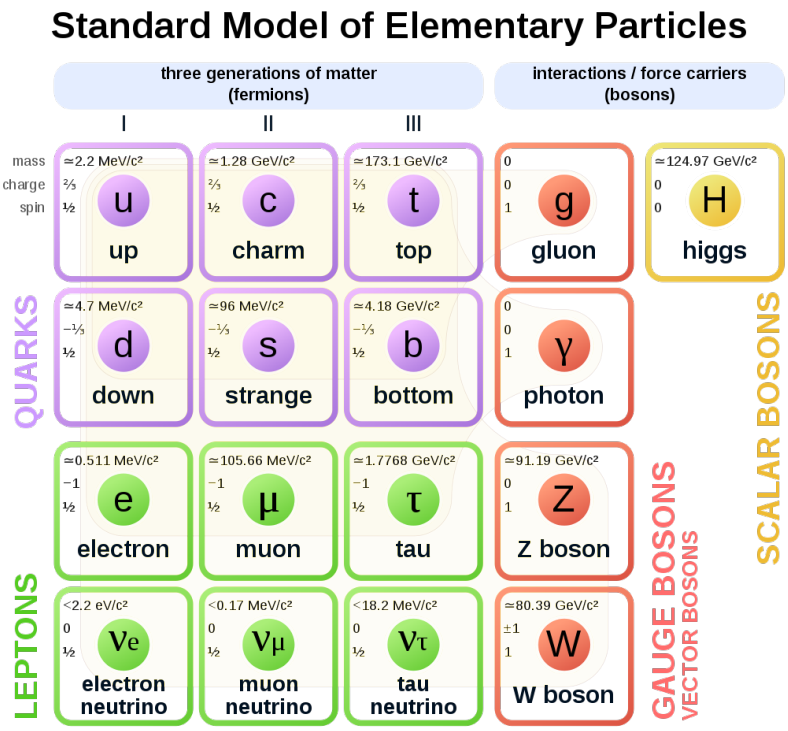


Figure 2.1: A common depiction of the fundamental particles in the Standard Model [22]. There are 12 fundamental fermions (quarks in purple and leptons in green) and 5 fundamental bosons (vector bosons in red and the Higgs in yellow). The three columns of fermions indicate the different generations of matter. The mass charge and spin values are also displayed for each particle. The neutrinos mass is now known not to be zero, but it has yet to be measured, so only upper limits are displayed.

Lagrangian of the SM dictates the dynamics and kinematics of the particles, each of which are described as a dynamical field that permeates space-time. In convention with the construction of most gauge field theories [3, 24] the SM is formulated first by assuming a set of symmetries. A Lagrangian can then be created which obeys those symmetries. Each global symmetry, in accordance with Noether's theorem [25], implies a conservation law of nature. Since the SM adheres to special relativity, the global Poincaré symmetry [26] is observed. This provides the Lagrangian with translational symmetry, rotational symmetry and inertial frame invariance, which in turn led to the conservation of momentum, space and angular momentum; conservation laws which exist in all relativistic quantum field theories. What uniquely describes the SM is the $\mathrm{SU}(3) \times \mathrm{SU}(2) \times \mathrm{U}(1)$ local gauge symmetry, of which each component roughly corresponds to a fundamental interaction.

The strongest of the three fundamental forces is aptly named the strong force. It is described through quantum chromodynamics (QCD) [27, 28], a Yang-Mills [24] gauge theory with $\mathrm{SU}(3)$ symmetry. It describes the interactions between the quarks and gluons in the SM, the latter of which are the vector bosons of the strong force. Particles can carry the QCD charge with one of three values which are labelled as red, blue or green. Only particles that have such a coloured charge will partake in

strong interactions, thus leptons are not affected by the QCD component of the SM Lagrangian. QCD exhibits two main properties, both of which have been experimentally verified. The first is colour confinement [29], whereby only colourless bound states of strongly charged particles are ever observed. This is because the energy required to separate two bounded coloured particles exceeds the energy required to spontaneously produce another quark-antiquark pair. The second property is that of asymptotic freedom [30], where strong interactions become asymptotically weaker as the distance between interacting particles decreases.

The remaining two forces in the SM are the electromagnetic force which is described by quantum electrodynamics (QED) [31], and the weak force which is described through the Glashow-Weinberg-Salam theory of electroweak processes [1–3]. Only particles carrying the electric charge will partake in electromagnetic interactions, which are mediated by the massless photon. Conversely, the weak force is the only interaction in the SM mediated by massive vector bosons: the W and Z bosons. The weak force has also been observed to violate parity conservation [32], as the W boson only couples to left-handed fermions (or right-handed antifermions). While the Z boson is not as exclusive, it also displays a similar preference.

At extremely high energies the electromagnetic and the weak forces become practically indistinguishable and are unified into the electroweak interaction. The electroweak sector of the SM Lagrangian is a Yang-Mills gauge theory with the symmetry group $\mathrm{SU}(2) \times \mathrm{U}(1)$. This unified theory has massless spin-1 vector bosons, but electroweak symmetry is broken by the Brout-Englert-Higgs mechanism [33–35]. This mechanism fixes many issues with electroweak theory particularly with how leptons and some of the bosons acquire mass. The addition of fermion mass terms into the electroweak Lagrangian is forbidden as they would not respect $\mathrm{SU}(2) \times \mathrm{U}(1)$ gauge invariance. Neither is it possible to add explicit mass terms for the individual gauge fields. In this theory a spin-0 field, called the Higgs field, permeates all space. The Higgs field also has a non-zero expectation value which causes spontaneous electroweak symmetry breaking, allowing the W and Z bosons to gain mass. The fermions then also acquire mass through Yukawa-type interactions with the Higgs field. Experimental evidence for this mechanism came in the form of the discovery of a spin-0 scalar boson in 2012 at the Large Hadron Collider [4, 36]. Further experimentation [37, 38] indicates that this is indeed the Higgs boson, an excitation of the Higgs field.

2.1.1 Shortcomings of the Standard Model

The SM has been an extremely successful theory. Figure 2.2 shows the accuracy with which it has predicted the cross-sections of many processes observed at the Large Hadron Collider. However, there are several shortcomings.

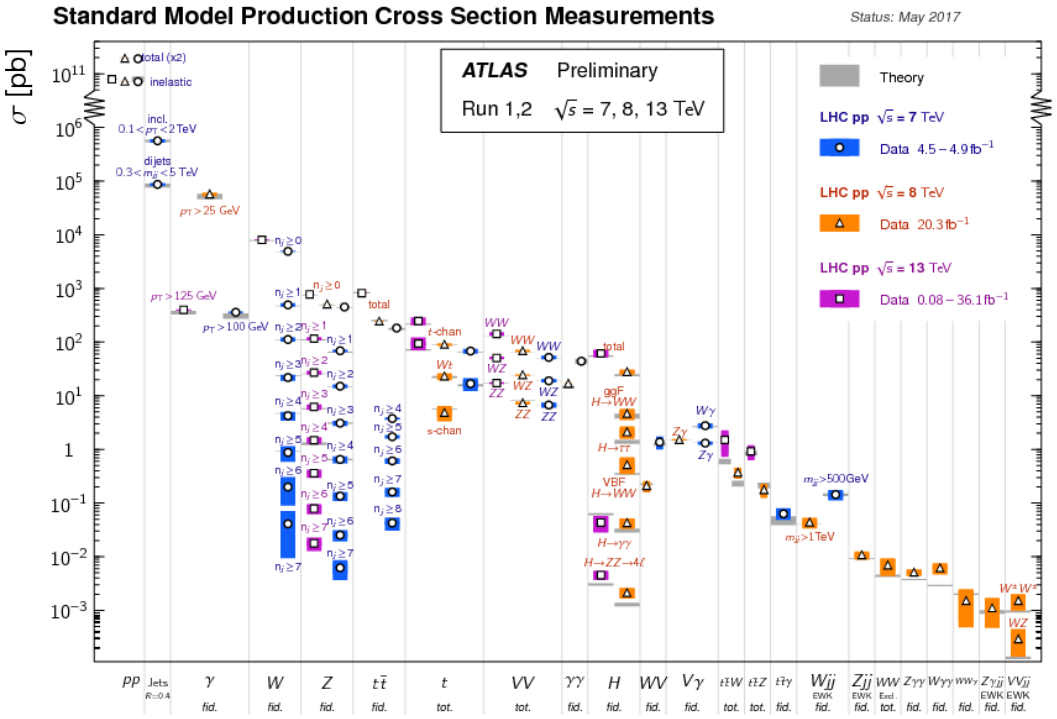


Figure 2.2: A summary of cross-section measurements in pp collisions at $\sqrt{s} = 7, 8, 13$ TeV for a variety of SM processes by the ATLAS collaboration compared to their theoretical predictions by the SM [39].

The baryon asymmetry [40] of the universe, which is the discrepancy between the amount of observed matter and antimatter, cannot be fully explained by the SM, as the observed violation of charge-conjugation-parity-symmetry (CP) in the weak interaction is insufficient to account for this discrepancy [41, 42]. Furthermore, while QCD theoretically also allows for CP violation, there is no experimental evidence for this. In fact, the strong force appears to preserve CP [43]. Why this is the case is known as the strong CP Problem [44, 45].

One of the most notable shortcomings of the SM is that it does not contain gravity. It is unclear if gravitation can be included at all using the current framework as there is no renormalisable QFT for gravitation. Gravity is around 10^{29} times weaker than the weak force, so gravitational quantum effects are only expected to become visible at the Planck scale (10^{19} GeV) [46]. This is 15 orders of magnitude greater than the energy range accessible to the LHC, and as such omitting gravity from the SM has not degraded its ability to predict phenomena at the microscopic scale. However, it does mean that no reliable theory exists for the very early universe.

This discrepancy in strength between gravitation and the weak force is related to another issue with the SM, called the Hierarchy Problem, which concerns the Higgs mechanism [46]. Since the Higgs boson couples to every massive particle, its mass receives radiative corrections from quantum loop processes. The corrections create a quadratically diverging sum and push the Higgs mass to infinity, or at least to the

chosen cut-off energy scale of the effective field theory [47], which is usually taken to be the Planck scale. The observed Higgs mass, and the energy scale of electroweak symmetry breaking, is obviously not this large, which requires an unnatural fine-tuning of the Higgs bare mass and its quantum corrections [20]. Alternatively, some other process outside the scope of the SM could be taking place.

Another failure of the SM is that while it claims to account for every fundamental particle, it lacks a viable candidate for Dark Matter [48], which makes up around 85% of all known matter in the universe. Evidence for Dark Matter first came from observations of the rotation curves of galaxies [49, 50], and has since become the favoured explanation in cosmology for many other observed phenomena, as opposed to modifying the current theories of gravity [51]. Dark Matter is assumed to be non-baryonic [52]. This is because it does not emit electromagnetic radiation, and large-scale structures like neutron stars or brown dwarfs are unable to account for all the Dark Matter observed in the universe [53, 54]. Dark Matter must be electrically neutral and cannot carry the colour charge. It is currently unclear if Dark Matter particles interact via the electroweak force at all. No candidate in the SM meets all these requirements and thus Dark Matter particles must exist outside the scope of established physics.

2.2 Supersymmetry

Supersymmetry (SUSY) [55–61] is currently one of the most popular extensions of the SM. In its minimal realisation (MSSM) [62–64] it postulates new fermionic partners to the fundamental bosons of the SM, and new bosonic partners to its fermions. It also introduces an additional Higgs doublet. If SUSY were an exact symmetry, then all supersymmetric particles (sparticles) would be nearly identical to their SM counterparts, with their spin being the only distinguishing feature. However, such sparticles should have been produced in abundance at the LHC by now. Since no supersymmetric partner for the electron has been observed, the symmetry must be broken. Like electroweak symmetry breaking, SUSY can be spontaneously broken at low energies, resulting in a hidden symmetry [57].

The additional particles of an MSSM model, as shown in Figure 2.3, include squarks, gluinos, and sleptons ($\tilde{l}$ and $\tilde{\nu}$). The SUSY partners for the Higgs and the electroweak fields mix to form the mass eigenstates known as charginos ($\tilde{\chi}^{\pm}$, $i = 1, 2$) and neutralinos ($\tilde{\chi}^0_j$, $j = 1, 2, 3, 4$). In MSSM theories, baryon and lepton number violation is prevented through the conservation of a new quantum number called R-parity [64]. Consequentially, sparticles may only be created or destroyed in pairs and the lightest supersymmetric particle (LSP) must be stable [65]. The LSP is typically assumed to be the lightest neutralino $\tilde{\chi}^0_1$ and it fulfils all currently known requirements of a Dark Matter particle [66, 67]. Furthermore, its mass may be of the order of 100 GeV, and thus it could be produced at the Large Hadron Collider. If Dark Matter is indeed

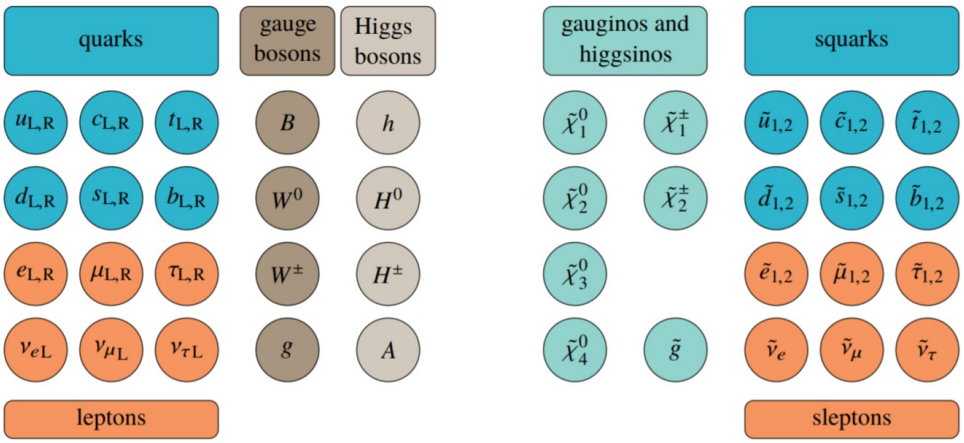


Figure 2.3: The particles present in MSSM theories include the original SM particles and their super-partners as well as an additional Higgs doublet. The sparticles are displayed according to their mass eigenstates [20].

comprised of weakly interacting massive particles (WIMPs) as many cosmologists believe, then many of these WIMPs would have been thermally produced after the Big Bang. To account for the correct abundance of dark matter observable today, these WIMPs require a self-annihilation cross section of around $3 \times 10^{-26}\,\mathrm{cm}^3\,\mathrm{s}^{-1}$. This value is roughly what is expected of an LSP in the 100 GeV mass range. This apparent coincidence is referred to as the WIMP miracle. And lends weight to the assumption that the LSP is a good WIMP particle.

The reason why SUSY is one of the most compelling SM extensions is that it elegantly answers many of the questions raised by the SM. SUSY provides a solution to the Hierarchy Problem [68–70], as the additional supersymmetric boson partners of the SM leptons cancel out the quantum corrections to the Higgs mass, preventing the diverging sum. It may explain the existence of Dark Matter through the LSP, and some SUSY models offer reasons for the asymmetry between matter and antimatter [40]. It can also enable the gauge coupling unification, in which the strong, weak and electromagnetic interactions are described by a single gauge group at the Grand Unification scale [71]. SUSY can even solve the strong CP Problem [72].

SUSY's strong appeal has inspired many attempts to verify the theory, however no supersymmetric particle has yet been discovered. Difficulties arise as the masses of the sparticles are free parameters and the entire parameter space must be searched to falsify the theory experimentally. The ATLAS experiment has ruled out several regions of the mass phase space, like the one shown in Figure 2.4.

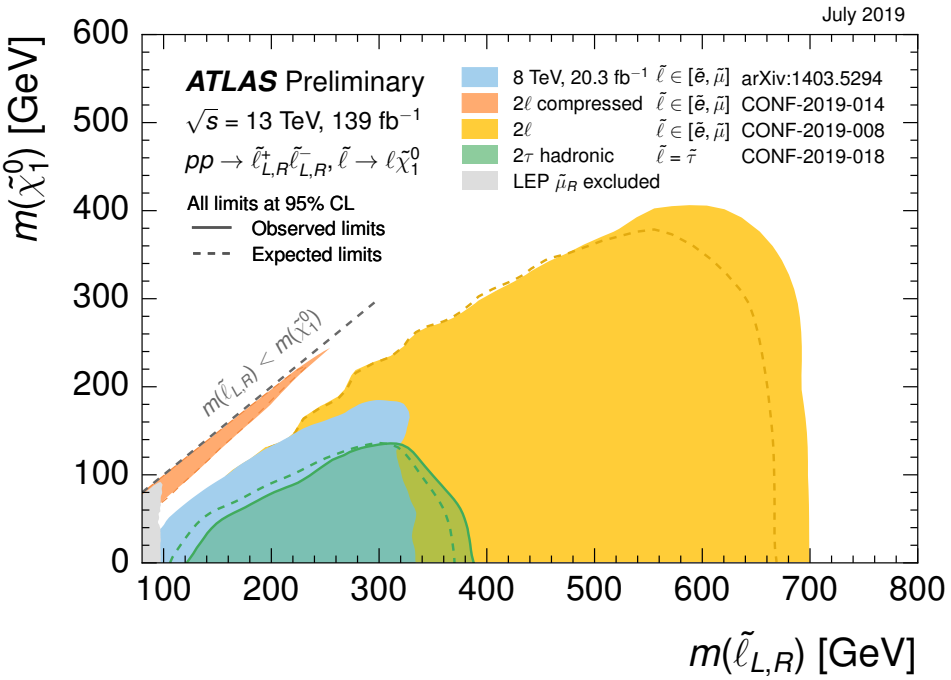


Figure 2.4: The exclusion region in the mass plane of the slepton and lightest neutralino at a 95% confidence-level based on previous searches at the ATLAS experiment [73].

Chapter 3

Deep Learning Methods

This chapter, like the one before it, attempts to outline the theories required to understand the work presented later on in this document. The concepts of machine learning, and particularly those behind ANNs, are presented. This chapter endeavours only to cover the principles directly utilised to develop the models in Chapter 9, and is not a complete description of the topic. Deep learning is one of the fastest growing fields in data science, and the following texts which were used as references for this chapter, offer a more thorough understanding [74–78].

3.1 Introduction to Machine Learning

Almost two centuries ago Augusta King, the Countess of Lovelace, thought about how one might utilise Charles Babagge's hypothetical invention; the Analytical Engine [79]. This was a highly advanced mechanical calculator and an early predecessor of the modern computer. Augusta realised that one might be able to write a set of instructions for the machine which would allow its function to shift from pure calculation to computing in the general sense as we know it today [80]. Due to this, Augusta King is regarded as the first computer programmer [81].

However, she disregarded the possibility that such a machine could think for itself and famously wrote,

> *" It is desirable to guard against the possibility of exaggerated ideas that might arise as to the powers of the Analytical Engine... The Analytical Engine has no pretensions whatever to originate anything. It can do whatever we know how to order it to perform. It can follow analysis, but it has no power of anticipating any analytical relations or truths."* [82]

Although this assessment holds true for the mechanical machine to which she was referring, modern computing has reached the point where it is almost trivial to construct a system which can automatically improve with experience. Such a system therefore contradicts Augusta King's statement that it should be limited to tasks which its creators "know how to order it to perform." Instead of explicitly writing down the steps a machine must take to perform a task, one can instead let the

machine teach itself how it is done. The machine develops its own method independent of its creators' knowledge of the problem at hand. The machine can anticipate relations and infer truths. This is the field of machine learning.

3.1.1 Definition

Machine learning can be loosely summarised as the automated practice of model building by extracting information from data to learn new tasks [78]. It has also been defined as the automated detection of meaningful patterns in data [76]. In both definitions the implication of the word "automated" is that, contrary to conventional coding, the algorithms and commands that make up the model are not explicitly programmed by humans. Instead, systems or machines are given statistical techniques to infer their own predictive algorithms based on examples.

The term "machine learning" was first phrased by Arthur Samuel in 1959 [83] when he was verifying that a computer program could employ such techniques to learn how to play a game of checkers. It was so successful that the computer reached a state where it would consistently defeat its creators. Despite its promise, computational limitations hindered its widespread use and up until 1985 there were almost no commercial applications [84]. As computer technology advanced in the 1990s, machine learning flourished. It has since become an umbrella term encompassing many different techniques and decision-making models which vary in how they learn. It is now one of the most exciting and fastest growing fields of data science. It has been implemented in a wide variety of areas with great success and has been responsible for many breakthroughs in computing, such as the rise of image recognition and computer vision [85].

There are several reasons why machine learning has been so successful. For some complex problems the sophistication of the required algorithms are much too intricate for direct construction. It is also used in cases where constant modification is needed after the model is fielded, such as speech recognition software which continuously adapts to the user. There are also many instances where automation is required simply because the datasets are too large. For example, the world's largest video sharing platform YouTube has around 82 years of video content uploaded every day [86]. These videos need to be identified, moderated and recommended to users at a rate which would be impossible for a human workforce. Thus Google, YouTube's parent company, has turned to using a deep neural network to handle most of these jobs [87].

3.1.2 Formalism

Since machine learning involves a computer system creating its own independent methodology, the field is seen as a significant branch of artificial intelligence. It is

therefore interesting from a philosophical point of view, as it requires an investigation into the principles which define intelligence. The sophistication and intelligence of a model is judged by how well it can learn to perform some preconceived task T. This requires some performance measure P, a quantification on how well T was executed. A machine is said to learn if P increases through experience E.

The Task, T

The task T is defined by how the machine learning algorithm should process an example of input data. The input example is a collection of features that have been quantitatively measured from some event or object that the algorithm needs to process. It is the standard to represent such an example of input data using the vector $\mathbf{x} \in \mathbb{R}^n$. The dimension of the input vector n describes the number of features or attributes a single example has. In general $\mathbf{x}$ could represent any complex or structured object. The result of running the machine learning algorithm can be expressed as a function acting on $\mathbf{x}$. The form of the function's output depends on the type of task T. Machine learning has proved to be particularly successful in tasks such as classification [85], regression [88], machine translation [89], transcription [90] and anomaly detection [91].

The Experience, E

In most cases of machine learning, experience is gained through the interaction of the model with a collection of examples called the training set $\mathcal{D}$. The manner of this interaction typically falls into one of three categories: supervised learning, unsupervised learning and reinforcement learning.

Unsupervised learning algorithms interact with a training set containing many examples of inputs $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^N$. Through this interaction the model learns useful properties of the training set. This might be the probability distribution(s) that generated the data or any underlying patterns, clusters, structures or anomalies. This type of machine learning is also referred to as self-organisation.

Reinforcement learning has the model interact perpetually with an environment. The model perceives this environment by being continuously fed sensory inputs in a time series. Based on these inputs $\mathbf{x}$ the model may act to change its environment. Learning takes place when, based on feedback from the model's actions, it is given occasional reward or punishment signals. This type of machine learning is the most analogous to classical conditioning. As the model interacts with the environment, it is attempting to recognise and anticipate what feedback it will receive. It can then behave in a manner to maximise the chances of getting a reward. Reinforcement learning has been successfully applied in several cases, such as teaching a robot to run [92], and even training an ANN to play complex team-orientated computer games which require high levels of coordination and communication [93].

Supervised learning is the most common type of machine learning method. It is also the method used to train the ANNs presented in this dissertation and therefore it is covered in greater detail.

These types of algorithms experience a training set made of couplets. For each input example $\mathbf{x}$ there is an associated desired output $\mathbf{y}$. The training set can be therefore represented by $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$. Similarly, to the input example, the desired output $\mathbf{y}$, also known as the target vector or response variable, can be any complexly structured object. A key idea here is that the training set $\mathcal{D}$ is a subset of a more complete set $\mathcal{C}$. The underlying assumption in supervised learning is that there exists some function f which maps from the input space $\mathcal{X}$ to the target space $\mathcal{Y}$, and therefore relates the elements of each couplet within some degree of error ϵ.

$$\exists(f : \mathcal{X} \to \mathcal{Y}) \text{ such that } f(\mathbf{x}) = \mathbf{y} + \epsilon \quad \forall(\mathbf{x}, \mathbf{y}) \in \mathcal{C} \tag{3.1}$$

It is the goal of supervised learning to create a model that approximates f with its own function and output $\hat{\mathbf{y}} = \hat{f}(\mathbf{x})$ by inferring relations using the training set only. The model's output $\hat{\mathbf{y}}$ must therefore belong to the same space $\mathcal{Y}$. Once this approximation is created, then it can be applied to new inputs whose targets are unknown. The expectation is that $\mathcal{D}$ and $\mathcal{C}$ contain roughly the same distributions and relations within the region of interest. If a large subset of the complete set is not represented in $\mathcal{D}$ this may bias the learning, as the wrong relations are being inferred.

The requirement that each example in the training set $\mathcal{D}$ comes with a desired output $\mathbf{y}$ is what differentiates supervised learning from unsupervised learning. Samples with these predefined targets are often called "labelled data". When $\mathbf{y}$ is categorical, the type of task required is called classification. If it is continuous, then the task is called regression. The learning algorithm attempts to estimate some mapping $f : \mathbb{R}^n \to \mathbb{R}^m$, where m corresponds to the dimension of the target vector. In Chapter 9, ANNs were trained using supervised machine learning techniques to produce a 2D vector: the missing transverse momentum of a pp collision. This is an example of regression where the function approximated was of the form $f : \mathbb{R}^n \to \mathbb{R}^2$.

The Performance Measure, P

The performance measure P is a quantified measure on how well the model is executing the required task T and is therefore highly specific. For problems such as classification, the performance measure might be the percentage of examples for which the model predicted the correct output. For regression, P could be a metric function $d : \mathbb{R}^m \times \mathbb{R}^m \to [0, \infty)$, defining the distance between the model's output and the target vectors, $d(\mathbf{y}, \hat{\mathbf{y}}) \geq 0$.

Since the performance measure should be a reflection on how well the model will work once deployed, it should always be evaluated on a set of data that is separate from the data used to train it. This orthogonal collection of examples is called the

evaluation set. The quality of a model is not defined by its performance on the training set due to a phenomenon called overfitting, which is explained in the context of ANNs in Section 3.6.

3.2 Deep Feed Forward Neural Networks

Artificial neural networks are one of the most widely known and employed systems in machine learning due to their robust approach at approximating multi-dimensional functions. These functions may have real-valued, discrete-valued or vector-valued outputs. Many textbooks use different notations and representations of neural networks. This dissertation follows the notation used in Reference [77].

An ANN is a machine learning model originally inspired by the structure of the biological brain. Simplified models of the brain show that it consists of many basic computational units called neurons which are interconnected via synapses. These connections form a complex communication network through which the brain can propagate information and carry out highly sophisticated tasks. ANNs are simply formal constructs attempting to mimic the same structure. Instead of electrochemical signals traversing the network, as in the biological case, ANNs propagate real valued numbers. Even though its original motivation was founded in biology, the brain contains many intricacies which are not modelled by ANNs. Conversely, there are many aspects of ANNs which are now known to be inconsistent with its natural counterpart. Despite this discrepancy, empirical evidence has already shown ANNs to be one of the most powerful models in machine learning.

In an ANN, neurons serve very simple mathematical routines, taking several inputs and producing a single real valued output. The output of one neuron may then be carried by a synapse to become the input of another, and so on. In some ANNs this information only travels in one direction. This basic model topology is referred to as the feed forward neural network. With a multitude of neurons, the network as a whole becomes a single, highly elaborate mapping. The complexity of this mapping is only limited by the size of the network. As explained in the previous section, the assumption behind supervised machine learning is that there exists some function f that links the couplets in the training set. Each neuron in the network has an associated real value called its bias b_i and each synapse has an associated real value called its weight w_i. The weights and biases of a neural network are its trainable parameters. They are represented by the vector $\theta = (w_1, ..., b_1, ...) \in \mathbb{R}^d$. They are randomly generated upon the network's initialisation and define the mapping $\hat{\mathbf{y}} = \hat{f}(\mathbf{x}; \theta)$. The ANN will then tune the trainable parameter values to produce a function best approximating f based on the training set. It can then make predictions on data it has never seen before. At its core, supervised learning with ANNs is highly analogous to curve fitting.

3.2.1 The Perceptron

The most common starting point for describing neural networks is the perceptron. A diagram of the perceptron is shown in Figure 3.1. This is a feed forward network with just a single neuron. A synapse connects each of the input features to the neuron, while multiplying them by a weight w_i. The neuron performs three functions: it sums up all its inputs, adds a bias term b, and then applies an activation function σ. Strictly speaking, the activation function used in perceptrons is the unit step function, but for other neural networks any non-linear function could be used. The output of a neuron is called its activation, and for the perceptron this is also the output of the entire network $\hat{y}$.

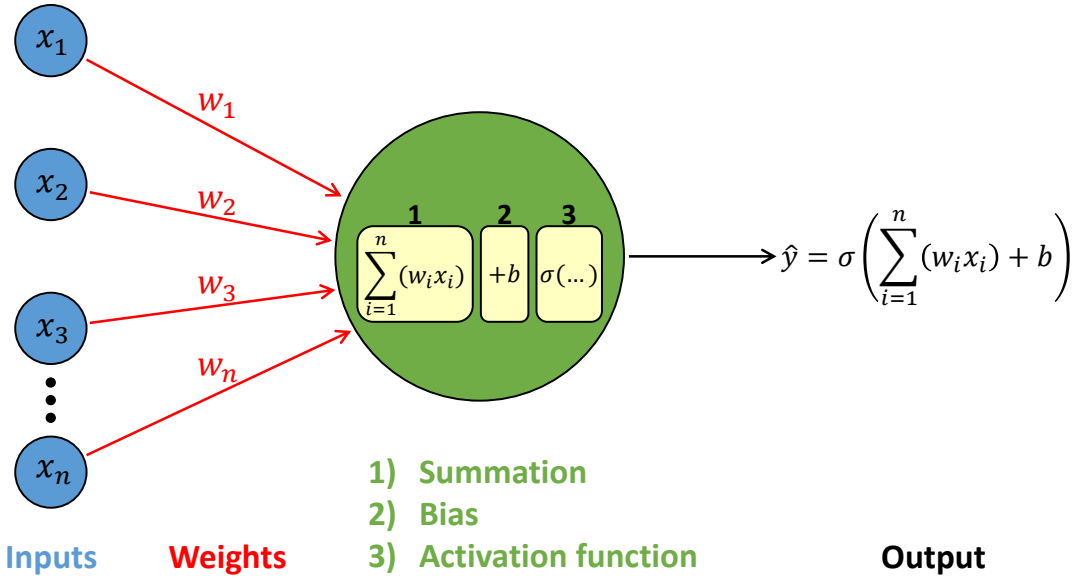


Figure 3.1: A diagram representing a perceptron, a neural network containing a single neuron (green). It is a feed forward network so information travels only in one direction, from left to right as indicated by the arrows.

The functional form of the model is expressed by:

$$\hat{y} = \sigma\left(\sum_{i=1}^{n}(w_i x_i) + b\right) = \sigma(\mathbf{w} \cdot \mathbf{x} + b) \tag{3.2}$$

Here σ is applied element-wise. As shown by Equation 3.2, the actual mathematics behind the perceptron is very simple and the space of functions it can approximate accurately is very limited. For example, it can only perform classification on linearly separable data. Since most real-world problems are much more complex, more sophisticated models are required.

3.2.2 The Multi-Layer Perceptron

By increasing the number of units, one is able to essentially compound many perceptrons and construct a more advanced neural network. A multi-layer perceptron, also known as a dense feed forward neural network, consists of these neurons arranged

in layers. A diagram of a multi-layer perceptron is shown in Figure 3.2. The leftmost layer is called the input layer and it matches the dimensions of $\mathbf{x}$ since it contains the input's features when it is passed to the network. As with the perceptron, nodes in the input layer are distinct as they do not contain bias terms nor activation functions. In the figure, the rightmost layer is called the output layer and similarly it will have the same dimensions as $\mathbf{y}$. In between these two, the network may have several intermediate hidden layers. The network in Figure 3.2 has three hidden layers.

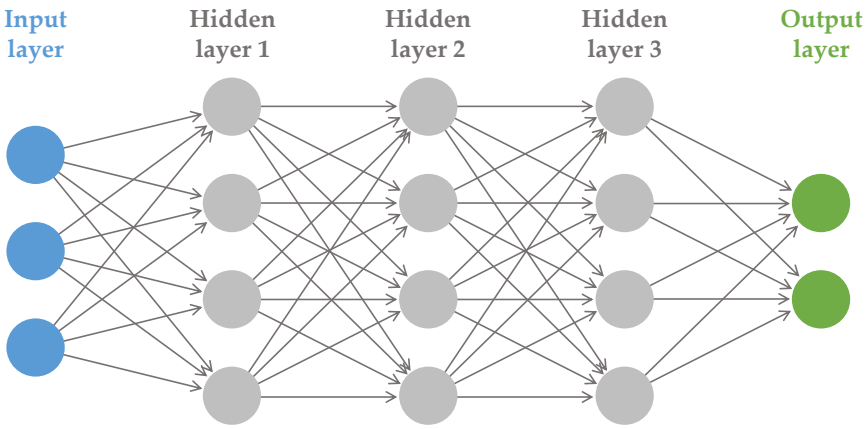


Figure 3.2: A representation of a multi-layer perceptron. The input layer (blue) has 3 neurons corresponding the dimension of vector $\mathbf{x}$. The output layer (green) has two neurons corresponding to the dimension of vector $\mathbf{y}$ and $\hat{\mathbf{y}}$. There are three hidden layers (grey) each with 4 neurons. This one network has 14 different bias values and 52 weights.

Each layer has an associated width given by the number of neurons it contains. A deep neural network contains at least one hidden layer, and the number of which is referred to as the network's depth. The network is referred to as "dense" since each hidden layer receives inputs from all neurons in the previous layer and outputs its activation to all neurons in the subsequent layer. This is also referred to as a fully connected neural network.

Layers are numbered according to their depth, starting at 0 with the input layer. The terms $b_j^{[l]}$ and $a_j^{[l]}$, as shown in Equation 3.3, denote the bias and the activation values of the j^{th} neuron in the l^{th} layer. Each synapse has its own weight, and $w_{jk}^{[l]}$ denotes the weight for the connection between the k^{th} neuron in the $(l-1)^{\text{th}}$ layer to the j^{th} neuron in the l^{th} layer. It is common practice for each neuron in the hidden layers to share the same activation function as this helps with computation. The activation function of the l^{th} layer is represented by $\sigma^{[l]}$.

The activation of the j^{th} neuron in layer l can be calculated using the same equation for the perceptron except the input features are replaced with the activations from the previous layer.

$$a_j^{[l]} = \sigma^{[l]}\left(\sum_k w_{jk}^{[l]} a_k^{[l-1]} + b_j^{[l]}\right). \tag{3.3}$$

This can be vectorised using $\mathbf{a}^{[l]} = (a_1^{[l]}, ..., a_p^{[l]})$ to represent the outputs of all neurons in layer l which has width p.

$$\mathbf{a}^{[l]} = \sigma^{[l]}(W^{[l]}\mathbf{a}^{[l-1]} + \mathbf{b}^{[l]}). \tag{3.4}$$

Here all the synapses between the two layers are represented by $W^{[l]}$, a $p \times q$ matrix where q is the width of the $(l-1)^{\text{th}}$ layer. Using this formalism, one can write the basic neural network model as a series of linear transformations interspaced with non-linear, element-wise activation functions. Equation 3.5 shows the functional form for a dense feed forward neural network with two hidden layers.

$$\begin{aligned}
\text{Input Layer} \qquad & \mathbf{a}^{[0]} = \mathbf{x} \\
\text{Hidden Layer 1} \qquad & \mathbf{a}^{[1]} = \sigma^{[1]}(W^{[1]}\mathbf{x} + \mathbf{b}^{[1]}) \\
\text{Hidden Layer 2} \qquad & \mathbf{a}^{[2]} = \sigma^{[2]}(W^{[2]}\sigma^{[1]}(W^{[1]}\mathbf{x} + \mathbf{b}^{[1]}) + \mathbf{b}^{[2]}) \\
\text{Output Layer} \qquad & \hat{\mathbf{y}} = \mathbf{a}^{[3]} = \sigma^{[3]}(W^{[3]}\sigma^{[2]}(W^{[2]}\sigma^{[1]}(W^{[1]}\mathbf{x} + \mathbf{b}^{[1]}) + \mathbf{b}^{[2]}) + \mathbf{b}^{[3]})
\end{aligned} \tag{3.5}$$

This representation shows why activation functions in the hidden layers are non-linear. If this were not the case then the entire model would be a composition of linear transformations, which itself could be rewritten as a single linear transformation. This would severely restrict the applicability of these models to only linear tasks. Equipped with non-linear activation functions, it can be shown that any mapping $f : \mathcal{X} \to \mathcal{Y}$ can be approximated by a large enough neural network (containing the correct set of parameters θ). This is known as the universal approximation theorem of neural networks [94] and is one of the reasons that they are so versatile. Some of the standard activation functions are discussed in Section 3.3. The universal approximation theorem holds true even for a network with a single hidden layer. However, there is significant advantage to be found in making networks deeper rather than wider. Increasing depth has been shown to be exponentially more valuable for approximation than width [95, 96]. Wider networks contain more parameters and are thus more prone to overfitting. Furthermore, deeper networks are able to develop concepts at various levels of abstraction.

3.2.3 Training a Neural Network

The definition of training a neural network can be summed up as searching for the best set of weights and biases that optimise the network's performance. This optimisation is defined in the context of minimising a cost function $C(\theta)$. It is also known as a loss or error function. This is slightly different from the performance measure P which is an external monitor of the intelligence of the network as a whole. Learning attempts to reduce a cost function $C(\theta)$ in the hope that doing so will also improve

P. For example, networks used for classification might use any number of loss functions during training, while the final performance of the model is defined by its classification accuracy.

The cost function is a continuous metric function dependent on network's parameters $C(\theta)$ which quantifies the distance between $\mathbf{y}$ and $\hat{\mathbf{y}}$. Simply put, it is just a description of the goodness of fit as seen in standard regression analysis, like the reduced chi-squared statistic. The training error of a model is the average cost over the entire training set.

$$C(\theta) = \frac{1}{N}\sum_{i=1}^{N} L(\hat{\mathbf{y}}_i, \mathbf{y}_i) = \frac{1}{N}\sum_{i=1}^{N} L(\hat{f}(\mathbf{x}_i; \theta), \mathbf{y}_i) \tag{3.6}$$

The cost evaluated on a single example is represented by L and its exact form depends on the model, the structure of $\mathbf{y}$ and the task T. Some cost functions, like cross entropy loss [97], are particularly useful for classification tasks. Some of the typical cost functions used in regression tasks are discussed in Section 3.5.

In normal statistical regression tasks, the goal is to find the parameters associated with the global minimum of the cost (chi-squared) across a set of data. However, in deep learning the network might contain millions of parameters. This makes it impossible to perform a thorough grid search or to minimise the cost analytically, so the global minimum is often never found. Instead, it is an acceptable solution to just find a local minimum that performs well enough. This is done using an iterative process called gradient descent.

Batch Gradient Descent

Networks are initialised at with random trainable parameters which are then modified in iterations. It is useful to visualise this minimisation technique by looking at the hypothesis space of possible weights and their associated C values. If a network only had two trainable parameters θ_1 and θ_2, then a 2D plane could represent the entire hypothesis space, as illustrated by Figure 3.3. The vertical axis in the figure shows the cost value C based on those two parameters. This cost hyper-surface summarises the desirability of every possible couplet of weight values. A low error implies a good hypothesis and a good function approximation. For standard gradient descent, also known as batch gradient descent, this cost value C is evaluated over the entire training set as described in Equation 3.6.

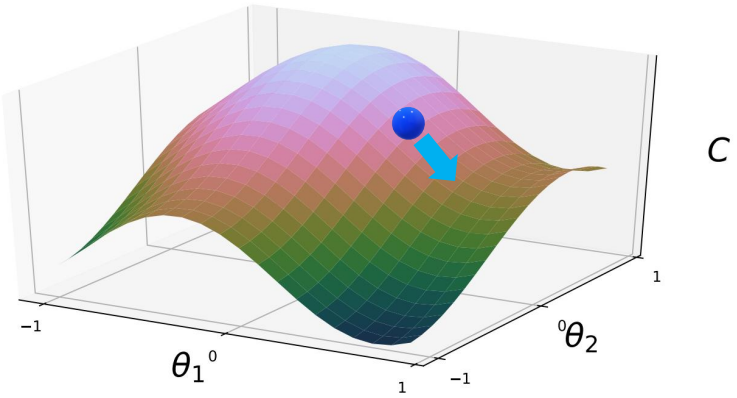


Figure 3.3: A visualisation of gradient descent. The cost hyper-surface C is defined for possible combinations of parameter values θ_1 and θ_2. The current state of the system is represented by the ball. To minimise the cost, the system moves a discrete amount in the steepest direction calculated using the gradient of the surface. This is analogous to the direction the ball would roll in a physical system. Repeated iterations of this step could converge on a local minimum.

With each iteration, the weight values are altered in the direction which produces the steepest descent along the error surface. This direction of steepest descent can be calculated using the gradient of the cost C with respect to the current individual parameters. This process continues until the system enters a minimum. Between each iteration the new cost and new gradients must be calculated. The update equation of gradient descent can therefore be written as:

$$\begin{aligned} \theta_1 &\leftarrow \theta_1 - \eta \frac{\partial C}{\partial \theta_1} \\ \theta_2 &\leftarrow \theta_2 - \eta \frac{\partial C}{\partial \theta_2}. \end{aligned} \tag{3.7}$$

Here η is a small positive hyperparameter called the learning rate. A hyperparameter is a configuration of the network that is set by the user before training. They are related to the network structure and are distinct from the trainable parameters θ which constantly change during learning. Methods for finding the optimal hyperparameters and model structure for a given task are discussed in Section 3.7.7. The learning rate dictates how large each step "downhill" should be. If the learning rate is too small, then it will take a very many iterations to reach a minimum. If the value is too big, then it is possible that the update step might jump over or bounce out of a minimum all together.

Equation 3.7 is written in vector format as

$$\theta \leftarrow \theta - \eta \nabla_\theta C(\theta). \tag{3.8}$$

Each of these iterations in batch gradient descent is referred to as an epoch. It can sometimes take hundreds of epochs for the network to converge on a minimum. Due

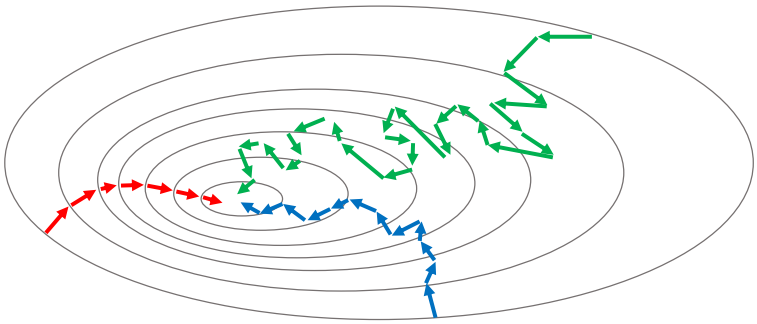


Figure 3.4: Each arrow represents a single update to the parameter vector using batch (red), mini-batch (blue) and stochastic (green) gradient descent. Using a smaller mini-batch size may give an early insight onto the rate of improvement of the model. Performing a single update step is also significantly quicker. Furthermore, it has been argued that the noisy updates allow the model to avoid local minima and premature convergence. Having a much larger batch size increases stability of the descent, and training can be better optimised on parallel architecture. However, larger batch sizes also mean training requires significantly more memory.

to the non-linearity of ANNs, the cost function is non-convex. This means there are more than one local minimum and it will be unlikely that gradient descent will find the global one. This is one of the most significant obstacles in training, but empirical evidence shows that ANNs are very effective at finding very good minima, even if there is no theoretical guarantee [98].

Calculating the gradients with respect to each weight and bias in the network used to be done through a process called backpropagation [99]. This is a very computationally costly method and most machine learning frameworks now calculate gradients using automatic differentiation, as discussed in Section 3.7.2. Batch gradient descent requires that the new cost and the new gradients are calculated for the whole training set to perform just one update. This can result in a very slow process and is sometimes unfeasible for large datasets which can't fit into memory. It also does not allow the model to be updated online. Therefore, different methods exist which vary in the amount of data used to compute the gradients, creating a trade-off between accuracy and the time taken to perform an update. The other methods are called stochastic gradient descent (SGD) and mini-batch gradient descent. A visualisation of these three methods is shown in Figure 3.4.

Stochastic Gradient Descent

As opposed to building the gradients over the entire training set, the other extreme approach is SGD. Here a parameter update is performed on each training couplet (x_i, y_i).

$$\theta \leftarrow \theta - \eta \nabla_\theta L(\hat{f}(\mathbf{x}_i; \theta), \mathbf{y}_i) \tag{3.9}$$

This significantly reduces the time between iterations. However, since SGD performs frequent updates to the parameters with a high variance, the descent fluctuates heavily as shown in Figure 3.4. This fluctuation may complicate the convergence, but it may also enable it to jump to new and potentially better local minimum or avoid saddle-points. SGD is performed by randomly shuffling the training set before passing them to the network in sequence. One epoch is completed after the network has processed every example in the set. It is then reshuffled and the process repeats.

Mini-Batch Gradient Descent

An intermediate form of these two methods is called mini-batch gradient descent and it is used by most deep learning optimisation algorithms. In this form, the data is still shuffled between each epoch, but it is then segmented into smaller orthogonal sets called mini-batches. Each mini-batch is called in turn and the batch form of gradient descent is performed.

$$\theta \leftarrow \theta - \frac{\eta}{M} \sum_{j=1}^{M} \nabla_\theta L(\hat{f}(\mathbf{x}_i; \theta), \mathbf{y}_i) \tag{3.10}$$

Here M is the number of examples in each mini-batch and is another network hyperparameter. After each mini-batch is processed, the epoch is complete, the data is reshuffled and re-partitioned. This is a more general form of gradient descent. If $M = N$ it is just normal batch gradient descent, and if $M = 1$ it is just SGD.

Mini-batch gradient descent is preferable since it reduces the variance of the parameter updates, leading to a more stable convergence than SGD. Furthermore, it can be used alongside batched forward propagation, covered in Section 3.7.1, which may mean it is sometimes quicker to perform than SGD.

3.2.4 Other Deep Learning Models

All deep learning models are based on ANNs, but the dense feed forward network is only a single subclass. Other, more complex structures exist which build upon the principles explained in this section. More detailed descriptions of the many deep learning models can be found in Reference [75], but a couple of the most common variants are mentioned here.

A special class of deep feed forward networks is the convolutional neural network [15]. These are specially designed for processing data with an underlying grid-like structure, such as an image. In place of the general matrix multiplication in Equation 3.4, these networks perform discrete convolutions using shared parameter values. These convolutions only take inputs from local receptive fields and thus learn to create feature maps which are applied across the whole example structure. They

are very good at detecting underlying patterns in data that have translational invariance. Convolutional neural networks have become the standard in computer vision [85].

Recurrent neural networks (RNNs) [100] include feedback connections allowing the model to use previous outputs when processing the next example. Information can be thought of as travelling backwards in the network and thus recurrent networks are distinct from feed forward networks. RNNs retain memories of previously studied examples and are thus well-suited for making predictions based off time series data, such as word prediction and language modelling [101].

3.3 Activation Functions

The choice of activation function plays a significant role in the network's ability to learn. There are many possible choices and there is unfortunately no clear best option. The optimal activation function varies from problem to problem, and like other hyperparameters it must be determined by trial and error, as explained in Section 3.7.7

Most activation functions share the same desirable characteristics. As previously mentioned, activation functions are non-linear as otherwise the entire network could be condensed into a single linear transformation, severely limiting its applicability. The other characteristic of activation functions is that they are differentiable. This allows gradients to be calculated and propagated throughout the network for parameter updates using gradient descent. It is also desirable that an activation function computes the identity at values close to zero. This assists learning when the weights are initialised with small random values. If this is not the case, special care must be used when initialising the weights [102]. Finally a good activation function is simple enough that, while it breaks linearity, it is also fast to compute, requires little memory, and has easily calculable derivatives.

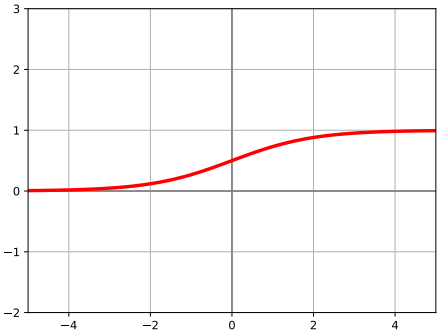
More than one type of activation function can be used in a network, but to simplify computation each neuron in the same layer typically applies the same function. The activation function in the output layer is usually distinct, as it must be chosen to match the form of the desired target y. This might mean that the final output function may need to be unbounded, set to an interval or positive definite. Since the application of the activation function takes place inside the neuron, it is common to call the neuron or unit by the name of the activation function it contains. For example, the term "exponential linear unit" refers to the function itself as well as a neuron containing it.

The following sections provide an overview of some of the common activation functions employed in deep learning. Their benefits and their drawbacks are discussed.

Sigmoid

The Sigmoid function is commonly referred to as the logistic function or the squashing function in literature. Before 2011 the standard logistic Sigmoid function and its scaled version, the hyperbolic tangent, were the most commonly used activation functions in deep learning [103]. They were inspired by probability theory and logistic regression.

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}$$



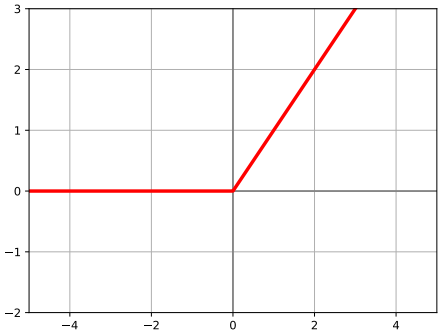
The Sigmoid function can be seen as a differential form of the perceptron's Heaviside function, allowing perceptrons to be trained using gradient descent. Their bounded output provided a mathematical description of a neuron which was either off or firing. The derivatives are easily calculated which made them attractive to early computer scientists who were more computationally constrained.

However, Sigmoid functions suffer from major drawbacks. The gradients tend to zero when the neurons saturate, and this can slow down learning. The non-zero centred output means that the weights associated with a Sigmoid neuron will all increase or all decrease together, which is very bad for convergence. Furthermore, the derivative of the function is bounded between 0 and 0.5. When networks contain multiple layers of Sigmoid neurons, the gradients diminish according to the chain-rule. Resulting in early layers of deeper networks that barely update. This is known as the vanishing gradient problem. All of these issues lead to very slow convergence and the Sigmoid neuron has since fallen out of favour. These days, Sigmoid neurons are only used in the final layer to clamp the network's output which can then be treated as a probability for logistic regression.

Rectified Linear Unit

The Rectified Linear Unit (ReLU) is the most popular activation function for deep neural networks. It was first proposed for use in machine learning in 2010 [104], and then in 2011 it was shown to improve the training speed, generalisation, and overall performance of deep neural networks which were then still using Sigmoid and hyperbolic tangent units [103].

$$\text{ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$



While many activation functions are designed to be continuously differentiable, the ReLU function contains a hard non-linearity and it is non-differentiable at zero. ReLU creates sparse representations in the network which has been found to be greatly suitable for sparse data. For example, in a randomly initialised network, only about 50% of hidden units would be activated, which can assist in information disentangling. It has been argued to have strong biological motivations [103]. Networks trained using the ReLU activation function encountered much fewer cases of the vanishing gradient problem that persisted in Sigmoid units. The simplicity of the function meant that both forward propagation and backpropagation across the entire network could be computed at much greater speeds, shortening training times.

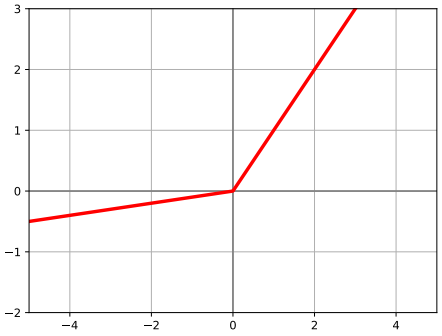
A significant drawback of ReLU is that neurons can be pushed into states where they become inactive for nearly all inputs. Since it is inactive, it contributes nothing to the final gradient and becomes perpetually stuck in this inactive state. This is known as the dying ReLU problem and it has been shown to get worse for deeper networks [105]. If many neurons become stuck in this state the model capacity is severely decreased. ReLU units also suffer from non-zero mean values which can still in some cases cause unstable gradients as discussed in 3.7.

Despite its drawbacks, the ReLU activation function is the most widely used in almost all areas of deep learning. It is used in applications from computer vision [85] to speech recognition [101]. While numerous activation functions have been proposed to replace ReLU, claiming to fix some aspects of its problems, none have yet managed to gain its widespread adoption. Many practitioners continue to use ReLU due to its simplicity and reliability.

Leaky, Parametric and Randomised ReLU

Specifically designed to counter the dying ReLU problem, the Leaky-ReLU (LReLU) function was proposed in 2013 [106]. This only changed the negative domain which in LReLU includes a small slope with gradient α. This is to keep the unit alive during the entire training process.

$$\text{LReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{if } x < 0 \end{cases}$$



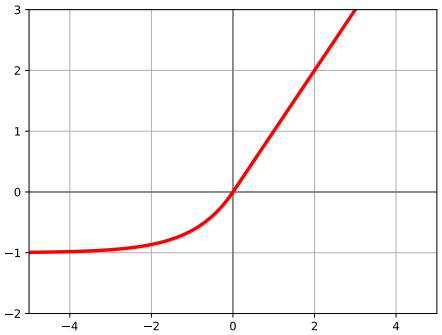
The LReLU function sacrifices the hard-zero sparsity for a gradient which allows it to be more robust during optimisation. It was initially tested on an automatic speech recognition dataset and achieved similar results to ReLU [106]. The value of α is taken to be 0.01 for most papers, but it is another hyperparameter to set for the network. One study showed that randomly sampling α for each unit at each training iteration could lead to better performance while also combating overfitting [107]. This is sometimes referred to as the Randomised Leaky-ReLU. Another paper investigated turning α into a trainable parameter [108], introducing the parametric ReLU function (PReLU). This meant that after an initial random initialisation, α would be modified with each iteration based on how it affects the cost function, just like the weights and biases of the network. Either the same α value is used for each neuron across the entire network or it can be unique for each unit. The initial study to use PReLU developed the first deep network to surpass human-level performance for visual recognition [108].

Since then, another study compared ReLU and its variants using various network architectures and datasets [107]. It found that the three derived forms, particularly PReLU, consistently outperformed standard ReLU.

Exponential Linear Unit

The Exponential Linear Unit (ELU) is another activation function based off ReLU. It was introduced in 2015 to increase the speed and stability of learning [109].

$$\text{ELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha(e^x - 1) & \text{if } x < 0 \end{cases}$$



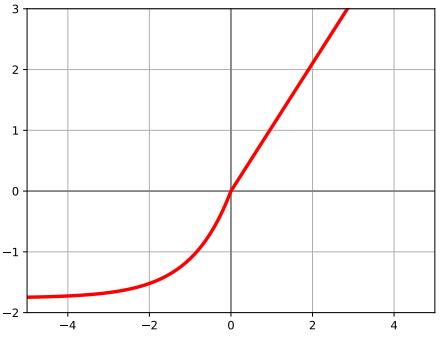
In most cases, the hyperparameter α is set to 1.0. A benefit of ELU is that, like ReLU, the positive portion has a constant gradient and therefore does not experience saturation, even at high values. It is very fast to calculate, and it alleviates the vanishing gradient problem. But unlike ReLU, ELU may output negative values which allows activations with means closer to zero, a desirable quality for networks as explained in Section 3.7. While LReLU and PReLU have negative values too, they do not ensure a noise-robust deactivation state. It is also close to the identity near zero and fully differentiable. In experiments, ELU led not only to faster learning, but also to significantly better generalisation performance on networks with more than 5 layers [110].

Scaled Exponential Linear Unit

The Scaled Exponential Linear Unit (SELU) is a slight modification to the ELU function and was introduced in 2017 [111]. The main motivation for its use was to improve the performance of dense feed forward networks. Two of the main obstacles facing networks with deep architectures is the unstable gradient problem and the bias shift covered in Section 3.7. Both of these issues can be countered with some form of normalisation which keep the activation of a single layer close to zero mean. However, SGD and stochastic forms of regularisation perturb normalisation efforts and lead to models with high variance in learning, not to mention that these normalisation steps are massively computation hungry.

Instead of normalising the outputs of the activation functions, SELU was developed as it intrinsically pushes outputs towards zero mean and unit convergence. The SELU function induces self-normalising properties like variance stabilisation which places an upper and lower bound on the variance, making vanishing and exploding gradients impossible [111]. The functional form is very similar to ELU.

$$\text{SELU}(x) = \begin{cases} \lambda x & \text{if } x \geq 0 \\ \lambda\alpha(e^x - \alpha) & \text{if } x < 0 \end{cases}$$



The important distinction to make is that λ and α are two fixed values. They are not trainable nor are they hyperparameters. The values were derived such that for standard scaled inputs, the output mean and variance of a layer would be zero and

one respectively. The values are:

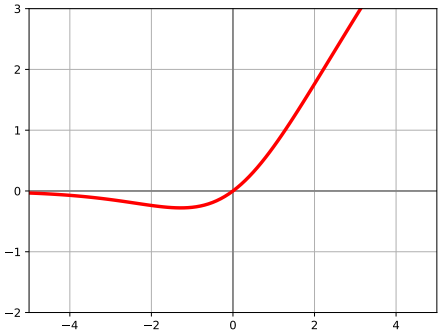
$$\begin{aligned} \lambda &= 1.0507009873554804934193349852946 \\ \alpha &= 1.6732632423543772848170429916717 \end{aligned} \tag{3.11}$$

The original paper showed that SELU performed slightly better than networks using both ReLU and batch normalisation, Section 3.7.6, for the MNIST and CIFAR10 datasets [111].

Swish

The Swish activation function is one of the first compound function used in neural networks. It is the combination of the Sigmoid function and the input function.

$$\text{Swish}(x) = \frac{x}{1 + e^{-x}}$$



Most of the aforementioned activation functions were specifically designed to possess qualities deemed important. Swish on the other hand was the product of an automatic search used to discover novel activation functions which showed strong empirical performance [112]. The search used reinforcement learning to generate a multitude of networks using different activation functions until it settled on the one with the best performance. The benefits of the Swish function were then empirically validated, showing that networks using Swish consistently outperformed otherwise identical networks using ReLU. It is the only non-monotonic function discussed in this dissertation and is described as a self-gated function.

Almost concurrently to the announcement of Swish, another research group showed that the same function led to improved performance with their deep learning models. They called it a Sigmoid weighted linear unit (SiLU) [113].

3.4 Gradient Descent Optimisation Algorithms

This section is primarily a summary of Reference [114] and Chapter 8 of Reference [75].

For vanilla stochastic, mini-batch, and batch forms of gradient descent, the parameter update is directly proportional to the learning rate and the gradient of the cost

function. The only difference between them lies in the number of examples used to calculate the gradient for each parameter iteration. As shown by Equation 3.10, all three forms can be generalised as mini-batch gradient descent with different sizes of M.

One of the challenges with these methods is choosing the proper learning rate η. A learning rate which is too small can lead to exceptionally long training sessions. Setting η to be too large might cause the loss function to fluctuate around a minimum, or in some cases diverge. When a stable learning rate is found, usually through trial and error, the same learning rate is applied across the entire network. This is an issue for sparse data, where one might want to perform larger updates for rarely occurring features.

An inevitability of training neural networks is that the gradient descent algorithm will converge to a sub-optimal local minimum when applied to a non-convex cost function. However, it has been argued that a deeper and more profound issue originates from the proliferation of saddle-points [115]. This issue is further exacerbated in high dimensional problems. When the iteration process in the parameter space is caught on such a saddle-point this can dramatically slow down learning, giving the illusion that it has reached a local minimum.

To overcome these issues, several different gradient descent optimisation algorithms have been developed. These optimisers modify the form of Equation 3.8. To simplify the equations below, $J(\theta)$ represents the average loss calculated over a mini-batch.

$$J(\theta) = \frac{1}{M} \sum_{j=1}^{M} L(\hat{f}(\mathbf{x}_i; \theta), \mathbf{y}_i) \tag{3.12}$$

Momentum

Gradient descent is a popular optimisation strategy, but it can sometimes be very slow [116]. It has trouble navigating ravines, areas in the parameter space where the cost hyper-surface rises more steeply in one dimension. This is very common around local minima. In such instances, gradient descent oscillates across the slopes of the ravine and only makes hesitant progress along the bottom towards the minimum. To accelerate this process, an additional momentum term is added to the algorithm [117, 118].

The momentum algorithm accumulates an exponentially decaying average of past gradients which helps move the system in the relevant direction. The hyperparameter γ determines the decay rate of previous gradients. The name of the method derives from the physical analogy of a particle moving through a potential field defined by $J(\theta)$. The field applies a force equal to the negative of its gradient, which changes the momentum of the particle. This can be numerically approximated by

the Euler method giving the following update equation:

$$\text{1) Accumulate velocity:} \quad \mathbf{v} \leftarrow \gamma\mathbf{v} - \eta\nabla_\theta J(\theta) \tag{3.13}$$

$$\text{2) Update parameters:} \quad \theta \leftarrow \theta + \mathbf{v} \tag{3.14}$$

In this interpretation the particle is assumed to have unit mass, so its momentum is equivalent to its velocity represented by $\mathbf{v}$. Here γ is commonly referred to as the momentum coefficient, but it is more analogous to the physical coefficient of viscous drag. It therefore lies within the range $(0, 1)$. Setting $\gamma = 1$ corresponds to no friction, but this is not ideal as the system must lose energy to settle within a minimum. It is common to use $\gamma = 0.9$.

The effect of momentum on the performance of the training process has been studied in-depth [119]. It can greatly decrease training times by increasing step-sizes along dimensions whose gradients continuously point in the same direction, while reducing those along dimensions whose gradients change directions frequently. It is therefore very effective in the face of high curvature and inconsistent or noisy gradients.

Nesterov Accelerated Gradient (NAG)

The potential issue with the momentum algorithm is its inability to slow down before reaching a minimum. This causes the system to shoot past the minimum and start moving up the slope of the parameter space on the other side. Nesterov momentum [120] or Nesterov Accelerated Gradient (NAG) was proposed to solve this problem by giving the system a small amount of prescience. This is done by calculating the gradient only after the current velocity is applied.

$$\text{1) Accumulate velocity:} \quad \mathbf{v} \leftarrow \gamma\mathbf{v} - \eta\nabla_\theta J(\theta + \gamma\mathbf{v}) \tag{3.15}$$

$$\text{2) Update parameter:} \quad \theta \leftarrow \theta + \mathbf{v} \tag{3.16}$$

The equation is constantly looking ahead of the system, and this anticipatory update allows the system to slow down before reaching a minimum. Nesterov momentum has been successfully employed in many different problems [121, 122] and has been shown in general to perform better than standard momentum techniques.

Adagrad

The Adagrad algorithm was the first widely used optimiser which employed adaptive learning rates [123]. It allows the learning rates to be tuned and adapted for each parameter. The learning rate is decreased for parameters associated with frequently occurring features, and updates parameters with larger steps if they are encountered by less training examples. This has been shown to be a great algorithm to employ when dealing with sparse datasets [124].

Adagrad adapts the learning rates of the trainable parameters by scaling each one based on a running total of its past squared gradients. For brevity, the vector $\mathbf{g}$ denotes the current gradient $\nabla_\theta J(\theta)$. These running totals of squared gradients are stored in the vector $\mathbf{s}$, which is zero at initialisation. The update equation therefore becomes:

$$\text{1) Compute gradient:} \quad \mathbf{g} \leftarrow \nabla_\theta J(\theta) \tag{3.17}$$

$$\text{2) Accumulate squared gradient:} \quad \mathbf{s} \leftarrow \mathbf{s} + \mathbf{g} \odot \mathbf{g} \tag{3.18}$$

$$\text{3) Update parameter:} \quad \theta \leftarrow \theta - \frac{\eta}{\sqrt{\mathbf{s} + \epsilon}} \odot \mathbf{g} \tag{3.19}$$

Here ϵ is a constant to prevent division by 0 and is of the order of 10^{-8}. The division and the square root in Equation 3.19 is applied element-wise.

An additional benefit of Adagrad is that it eliminates the need to manually tune the learning rate. It is standard implementation to use $\eta = 0.01$ for most problems. However, the greatest weakness of using the Adagrad formula is that the learning rate is always in a constant state of decay. Since each term added to g is positive, the learning rate soon becomes infinitesimally small, at which point the model is no longer able to acquire additional knowledge.

RMSProp

RMSProp is an unpublished, adaptive learning rate optimiser which was proposed by Geoff Hinton in his series of machine learning lectures [125]. It is an extension of the principles introduced by the Adagrad algorithm and was designed to solve the issue of Adagrad's monotonically decaying learning rate. Instead of accumulating the squared gradients from the start of learning, RMSProp uses an exponentially decaying average. The decay rate of this average is represented by $\gamma \in [0, 1)$, though it is not to be confused with the momentum coefficient. The update equation is given by:

$$\text{1) Compute gradient:} \quad \mathbf{g} \leftarrow \nabla_\theta J(\theta) \tag{3.20}$$

$$\text{2) Accumulate squared gradient:} \quad \mathbf{s} \leftarrow \gamma\mathbf{s} + (1 - \gamma)(\mathbf{g} \odot \mathbf{g}) \tag{3.21}$$

$$\text{3) Update parameter:} \quad \theta \leftarrow \theta - \frac{\eta}{\sqrt{\mathbf{s} + \epsilon}} \odot \mathbf{g} \tag{3.22}$$

It has been shown that RMSProp works better in a non-convex setting than Adagrad.

Adadelta

Adadelta [126] is a very similar algorithm to RMSProp, though both were developed independently and around the same time. Adadelta was designed to fix the same issues with Adagrad, but its creators further noted that the units in the update

equation of gradient descent do not match the same units of the parameters θ .

$$\text{units of } \Delta\theta \propto \text{units of } \mathbf{g} \propto \text{units of } \nabla_\theta J \propto \frac{1}{\text{units of } \theta} \tag{3.23}$$

This is assuming that the loss function is itself dimensionless. To solve this issue, the Adadelta paper suggests that another exponentially decaying average is defined $\mathbf{d}$, this time looking at the previous values of the parameter updates $\Delta\theta$. This value is used to replace the learning rate and a single iteration of the modified update equation can be written as follows.

$$\text{1) Compute gradient:} \quad \mathbf{g} \leftarrow \nabla_\theta J(\theta) \tag{3.24}$$

$$\text{2) Accumulate squared gradient:} \quad \mathbf{s} \leftarrow \gamma\mathbf{s} + (1-\gamma)(\mathbf{g} \odot \mathbf{g}) \tag{3.25}$$

$$\text{3) Accumulate squared update:} \quad \mathbf{d} \leftarrow \gamma\mathbf{d} + (1-\gamma)(\Delta\theta \odot \Delta\theta) \tag{3.26}$$

$$\text{4) Compute parameter update:} \quad \Delta\theta \leftarrow -\frac{\sqrt{\mathbf{d}+\epsilon}}{\sqrt{\mathbf{s}+\epsilon}} \tag{3.27}$$

$$\text{5) Apply update:} \quad \theta \leftarrow \theta + \Delta\theta \tag{3.28}$$

Adadelta therefore eliminates η from the update equation, so one does not even need to set the learning rate.

Adam

While Adadelta and RMSProp have shown great improvements over Adagrad, they do not utilise the concept of momentum. Adaptive Moment Estimation (Adam) is yet another adaptive learning rate optimisation algorithm which uses a decaying average of both the first-order $\mathbf{m}$ and second-order $\mathbf{s}$ moments of the gradients. It can therefore be seen as a combination of both RMSProp and momentum [127]. Momentum in Adam is directly incorporated as an estimate of the first-order moment of the gradient. However, Adam includes bias corrections to the estimates of the moments. This is because the authors noted that since the moments were initialised as zeros, this created a bias which was especially noticeable during the initial time steps or when decay rates were small. The creators of Adam counteract these biases by computing correction estimates. These correction factors were dependent on the total number of iterations t.

The full Adam update equation is given by:

$$\text{1) Compute gradient:} \quad \mathbf{g} \leftarrow \nabla_\theta J(\theta) \tag{3.29}$$

$$\text{2) Update biased first moment estimate:} \quad \mathbf{m} \leftarrow \gamma_1 \mathbf{m} + (1-\gamma_1)(\mathbf{g}) \tag{3.30}$$

$$\text{3) Update biased second moment estimate:} \quad \mathbf{s} \leftarrow \gamma_2 \mathbf{s} + (1-\gamma_2)(\mathbf{g} \odot \mathbf{g}) \tag{3.31}$$

$$\text{3) Correct bias in first moment:} \quad \hat{\mathbf{m}} \leftarrow \frac{\mathbf{m}}{1-\gamma_1^t} \tag{3.32}$$

$$\text{3) Correct bias in second moment:} \quad \hat{\mathbf{s}} \leftarrow \frac{\mathbf{s}}{1-\gamma_2^t} \tag{3.33}$$

$$\text{5) Update parameter:} \quad \theta \leftarrow \theta - \eta \frac{\hat{\mathbf{m}}}{\sqrt{\hat{\mathbf{s}}} + \epsilon} \tag{3.34}$$

The original paper suggested default values of $\gamma_1 = 0.9$, $\gamma_2 = 0.999$ and $\epsilon = 10^{-8}$. They also show that Adam performs favourably to other optimiser methods and requires very little modification of the initial learning rate. Its averaging over past gradients corresponds to a large velocity that makes it resistant to falling into small regions. It tends to prefer flat minima which generalise well. In this regard Adam has been described as behaving like a heavy ball with friction [128]. Adam has become one of the most popular optimisation methods for deep learning especially in cases where speed is a priority.

AdaMax

The AdaMax optimiser is a modification of Adam which the authors proposed in the same paper [127]. In Adam, the second moment vector $\mathbf{s}$ is updated using the current gradient squared as shown in Equation 3.31. This can be rewritten as taking the value of the l_2 norm on the gradient. The authors found that testing with higher order normalisations resulted in very unstable algorithms, except when using the infinity norm l_∞. The infinity norm acting on a vector is equal to its maximum value. AdaMax replaces the second moment estimate with the infinity norm of the gradient. This updated moment is not as suggestible to a bias of zero and thus receives no correction.

$$\text{1) Compute gradient:} \quad \mathbf{g} \leftarrow \nabla_\theta J(\theta) \tag{3.35}$$

$$\text{2) Update biased first moment estimate:} \quad \mathbf{m} \leftarrow \gamma_1 \mathbf{m} + (1-\gamma_1)(\mathbf{g}) \tag{3.36}$$

$$\text{3) Update infinity norm gradient estimate:} \quad \mathbf{s} \leftarrow \max(\gamma_2 \mathbf{s}, \mathbf{g}) \tag{3.37}$$

$$\text{3) Correct bias in first moment:} \quad \hat{\mathbf{m}} \leftarrow \frac{\mathbf{m}}{1-\gamma_1^t} \tag{3.38}$$

$$\text{5) Update parameter:} \quad \theta \leftarrow \theta - \eta \frac{\hat{\mathbf{m}}}{\mathbf{s}} \tag{3.39}$$

It was argued that AdaMax was more stable than Adam and thus more suitable for sparsely updated parameters.

3.5 Loss Functions for Regression Tasks

The loss or cost function is a quantification of how far off the model's prediction $\hat{\mathbf{y}} = (\hat{y}_1, ...\hat{y}_m)$ was from the target vector $\mathbf{y} = (y_1, ...y_m)$. The gradients of this function are used to update the parameters of the network and therefore its form has a significant impact on how the network learns, what its priorities are, and its final performance. There is no single loss function that is the optimal choice for all tasks. Some factors to consider are the presence of noise and outliers in the dataset, the choice of machine learning algorithm, the time efficiency for calculating the gradients, and the confidence of predictions. Loss functions are also broadly categorised into those good for regression or classification tasks, though with some overlap.

This section covers three loss functions commonly used in regression tasks and were the ones chosen to train the networks presented in Chapter 9. They are shown in Figure 3.5.

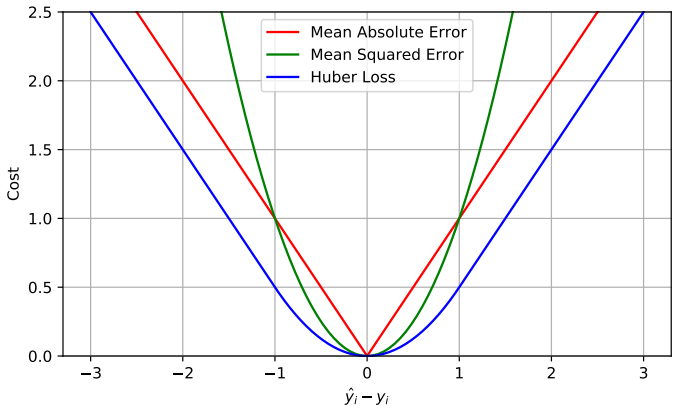


Figure 3.5: The loss functions corresponding to mean absolute error (red), mean square error (green) and Huber loss (blue).

The mean square error (MSE), also known as the quadratic loss or L^2 loss, is the most commonly used regression loss function. MSE is the sum of the squared distances between the target variable elements and the model predictions.

$$\text{MSE}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \tag{3.40}$$

MSE is useful for analysing the performance of linear regression models, as it allows one to distinguish between errors caused by systematic and stochastic sources. Gradient descent using MSE produces an unbiased estimator of the arithmetic mean. However, squaring each term means that large errors are heavily weighted, which

may be undesirable in many applications. This is particularly the case in regression tasks with large outliers or heavy-tailed distributions.

To produce more robust regression models, the mean absolute error (MAE), also known as L^1 loss, is preferable. This loss function leads to unbiased estimator of the geometric median. MAE is calculated using the absolute differences between the target and predicted vector elements.

$$\text{MAE}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{m} \sum_{i=1}^{m} |\hat{y}_i - y_i| \tag{3.41}$$

The MAE loss is useful if the training data is corrupted with outliers. However, the MAE gradient is constant and does not vanish as the error gets small. A training example which is nearly perfectly modelled will cause the network to update by the same magnitude as an example which is very poorly modelled.

The Huber loss function, also known as smooth-L^1 loss, was designed to combine the advantages of the MAE and the MSE. Huber loss is calculated elementwise and is defined by the piecewise function,

$$\text{Huber}(\hat{y}_i, y_i) = \begin{cases} \frac{1}{2}(\hat{y}_i - y_i)^2 & \text{if } |\hat{y}_i - y_i| < 1 \\ |\hat{y}_i - y_i| - \frac{1}{2} & \text{otherwise} \end{cases} \tag{3.42}$$

This value is then averaged for all elements in the target and output vectors. Huber loss is less sensitive to outliers in data than MSE, but is better than the MAE for noisy data as the gradients tend to zero when the error is relatively small. It is therefore well suited for most problems.

3.6 Regularisation

The training process described in Section 3.2.3 was defined as finding the parameters of the network that minimise the training error. However, a key challenge in machine learning is to create a model that can generalise and perform well on new and previously unseen inputs. This is why the performance measure as described in Section 3.1.2 is calculated on an orthogonal set of examples called the evaluation set. The cost measured on the evaluation set is referred to as the evaluation error or generalisation error. A good performing model is one that, through minimising the training error, also leads to a very small generalisation error.

For neural networks, this is a significant distinction as they are particularly susceptible to overfitting. Overfitting is not limited to machine learning and can be used to describe any statistical model. It occurs when a model learns specific details about its training data that do not represent general properties of the population. An overfit model might achieve very low training error but would fail to fit additional data or predict future observations accurately. The easiest method to reduce overfitting

would be to increase the size of the training set. Unfortunately, this is not always feasible due to limitations in available data, or in the context of this project, limited computing resources to create large simulated datasets. For a fixed training set size, a model's susceptibility to overfitting is correlated to the number of free parameters it contains. Since deep neural networks can contain millions or even billions of trainable (free) parameters, they are very prone to overfitting.

Since many tasks require highly complex models, simply reducing the size of the network and therefore the number of free parameters is undesirable. In many deep learning tasks, the model producing the best generalisation error is a very large and complex one that just has been regularised. Regularisation can be described as any strategy designed to reduce the generalisation error of a model, even if that means sacrificing accuracy on the training set. The following section describes the regularisation methods used in this analysis, which are some of the most common techniques, but a more extensive list can be found in Chapter 7 of Reference [75]. These methods are also not exclusive and are sometimes used in combination with one another.

3.6.1 Early Stopping and the Holdout Method

Cross-validation refers to any technique used to estimate the generalisation capabilities of a statistical model and the holdout method is the simplest form of cross-validation. Here an additional dataset is introduced called the testing set. Learning takes place exclusively on the training set as before, but between each epoch the model is asked to predict output values for data in the testing set. The cost on the testing set is called the testing error is calculated. During descent, the training error is expected to demonstrate a downward trend. For very large models which possess sufficient representational capacity to overfit the task, the testing error initially decreases, but at some point begins to rise again, as shown in Figure 3.6. The increase in testing error indicates the stage where the model is learning relations specific only to the training set.

In the holdout method, the model parameters are saved each iteration only if they led to a better testing accuracy. Early stopping is the strategy whereby training is terminated if there has been no improvement over the best recorded testing error for some pre-specified number of epochs. This is known as the patience. The patience should always be fairly large, as a noisy gradient descent encountering a saddle-point might give the illusion that the testing accuracy has saturated. If training is terminated too early, this will result in non-optimal performance. Upon termination, the parameters that produced the minimum testing error are returned.

Early stopping is one of the most commonly used methods of regularisation due to its effectiveness and simplicity. However, the cross-validation may depend heavily on which examples end up in the training set and which end up in the testing set.

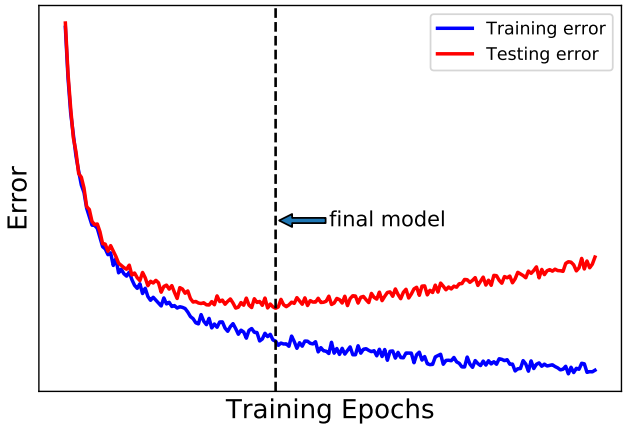


Figure 3.6: An illustration of the typical evolution of the errors associated with a neural network. Both training and testing errors decrease initially, however at a certain point overfitting becomes dominant and the testing error begins to rise. Early stopping is the process where training is terminated and the final model is the one which corresponds to the best testing accuracy, as indicated by the vertical line.

The testing error may be significantly different depending on how the division is made. The testing set is also orthogonal to the evaluation set as it serves a different purpose. So, the three orthogonal sets and their uses are as follows. The training set is used to directly update the parameters of the model. During this process the model's accuracy is measured on the testing set to monitor overfitting and to perform early stopping. Only once training has ended, and the final model has been produced, is the evaluation set used to gauge its performance and generalisation error.

Parameter Norm Penalties

Various parameter norm penalties have been used to regularise regression models for decades prior to the rise of deep learning. These techniques stem from the idea that individual weights should not carry too much influence on the model's output. This restriction is achieved by adding a penalty $\Omega(\theta)$ to the loss function C. The regularised loss function is denoted by $\widetilde{C}$.

$$\widetilde{C} = C(\theta) + \alpha\Omega(\theta) \tag{3.43}$$

Here $\alpha \in [0, \text{inf})$ is a hyperparameter that scales the relative contribution of the penalty term compared to the standard cost function. In neural networks the parameter norm penalty only penalises the weights $\mathbf{w}$ of the network and leaves the biases unaffected. Applying regularisation to the biases has been shown to lead to a significant amount of underfitting. This form of normalisation can be thought of as optimising the cost function while placing constraints on the weights of the network.

L^2-Parameter Regularisation

The norm penalty known as L^2 regularisation is one of the most commonly used methods in deep learning. It is also referred to as ridge-regression or weight-decay. This method pushes the weights of a network towards smaller values by adding a penalty term based on the L^2 norm of the weight vector.

$$\widetilde{C} = C(\theta) + \alpha \left(\frac{1}{2} \mathbf{w}^\intercal \mathbf{w} \right) \tag{3.44}$$

The name weight-decay becomes evident when considering how the normalisation affects the gradient descent update equation.

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_\theta \widetilde{C}(\theta) = (1 - \eta\alpha)\mathbf{w} - \eta \nabla_\theta C(\theta) \tag{3.45}$$

The addition of the L^2 penalty results in multiplicative shrinks of the weight vector by a constant factor of $(1 - \eta\alpha)$ before performing the usual update. This type of normalisation heavily penalises larger weight values, keeping weights small and diffuse.

L^1-Parameter Regularisation

Another parameter penalty can be constructed using the L^1 norm. This method is also referred to as lasso regression.

$$\widetilde{C} = C(\theta) + \alpha \left(\sum_i |w_i| \right) \tag{3.46}$$

The update equation therefore becomes:

$$\mathbf{w} \leftarrow \mathbf{w} - \alpha \text{sign}(\mathbf{w}) - \eta \nabla_\theta C(\theta) \tag{3.47}$$

Here $\text{sign}(\mathbf{w})$ is applied element wise. Before the gradients are applied, all positive weights decrease and all negative weights increase by a constant value of α. If the gradients of the standard cost function are too small to compete with this effect, which would be the case for unimportant parameters, the weights are reduced to zero. This results in very sparse networks which are efficient at feature selection. It can be seen as an automatic trimming of the network.

3.6.2 Dropout

Dropout [129] is a straightforward but very powerful regularisation technique. It involves the process of ignoring or "dropping out" a random set of neurons during training time. Dropout can be applied to individual layers of the network. The probability for each neuron to be dropped is p. A dropped neuron is effectively removed from the network along with all its incoming and outgoing connections. To

compensate for the reduction of signals in the network, the surviving neurons have their activations scaled by $1/(1-p)$ during training. At the start of the next descent step, all neurons are reactivated, and a different random selection is chosen to be dropped. The final network used for evaluation has all neurons active. Compared to other regularisation techniques, dropout is extremely computationally inexpensive. It has also been shown to work very well with nearly every deep learning model.

Several explanations have been proposed as to why dropout is so beneficial. One claim is that it prevents units and their weights from co-adapting. Later layers learn how to cope if incomplete information is provided. Dropout therefore forces each hidden unit to not only be a good feature, but a feature that is good in many contexts. The randomness of dropout also injects noise into the network, and it has been proposed that this is what assists in regularisation. Dropout also simulates a sparse network which in-turn encourages sparse representations. Dropout has also been considered a form of ensembling. During training, each mini-batch of data encounters a slightly different network. The final model can therefore be seen as an averaging of multiple stochastic decision methods. More recently, dropout has been seen to be an approximate form of variational inference in Bayesian neural networks [130].

3.7 Further Training Optimisations

This section covers some of the common techniques used to improve the way that neural networks learn. These techniques were used to train the models presented in Chapter 9.

Deep fully connected neural networks are no longer the most common type of deep learning model. Most networks for computer vision contain convolutional layers. Recurrent networks are the optimal choice for any problems that require sequencing. Outside these tasks, deep learning is often outperformed by boosted decision trees (BDTs), random forests or support vector machines. Dense feed forward networks remain relatively shallow, around 4-5 layers [111]. While a deeper network would allow for more abstract representations of the input, fully connected networks with many layers are not performing as well as many would have hoped. This is primarily due to the many difficulties training a deep and dense network.

One of the main issues is the unstable gradient problem. Depending on the scale of the activations, this could either manifest in a vanishing or an exploding gradient. The gradients in the early layers of the network are products of terms from all later layers. So the deeper the network, the more intrinsically unstable this problem becomes. The only way to negate this, is to have all the terms balance each other out. This can be done by ensuring that the activations of each layer have zero mean and approximately unit variance. Some of the activation functions listed in Section 3.3 attempt to do just that.

The normalisation of activations not only helps with the unstable gradient problem but also improves the capabilities of the model. When a single unit has a non-zero mean activation across a dataset, and this is not cancelled out by other units in the same layer, this acts as a bias for the next layer. During training this would produce a bias shift. Fisher optimal learning, which implies learning using the natural gradient [131], would correct for this bias. Activations with zero mean reduce the bias shift, bringing the standard gradient closer to the natural gradient which in turn speeds up learning. Some of the techniques described in this section have the specific goal of normalising the activations of the network.

3.7.1 Batched Propagation

Passing an entire mini-batch of training examples through a network can be done in a single step, rather than one at a time like Equation 3.4 implies. This greatly improves the computation time, especially on hardware optimised for large tensor operations. The mini-batch of inputs $\{\mathbf{x}_1, ..., \mathbf{x}_m\}$ is contained in an $M \times n$ matrix X, where each row represents a different input example. It can be shown that the activation of a single layer in a fully connected neural network can be calculated for the entire batch using the following equation.

$$\mathbf{A}^{[l]} = \sigma^{[l]}(\mathbf{A}^{[l-1]}\mathbf{W}^{\intercal} + \mathbf{B}^{[l]}). \tag{3.48}$$

In this formalism, each row of the matrices represents a single training example. The bias matrix $\mathbf{B}^{[l]}$ is created by duplicating the layer's bias terms for each row. This format best represents how the data is processed mathematically by a computer. It is also why mini-batch gradient descent requires more memory than SGD, but can lead to lower epoch times.

3.7.2 Specialised Hardware and Software

Machine learning software packages and libraries, such as PyTorch [132] and TensorFlow [133], are necessary these days for deep learning applications. On top of providing simple methods to construct network features, each package is also able to perform automatic differentiation. In brief, automatic differentiation is a feature which records the complete history of all operations applied to selected variables. While this does result in heavy memory usage, it allows for the rapid calculation of gradients necessary for deep learning with gradient descent. Before the use of automatic differentiation, gradient descent relied on the backpropagation algorithm. Neural networks were severely limited in the type of layers they could contain and training times were much longer.

Graphical processing units (GPUs) are a type of specialised hardware found in most personal computers. They were originally developed to handle three-dimensional

(3D) graphical rendering in video games. These jobs require many tensor operations which are highly parallelisable. GPUs contain many simple cores which allow computation through thousands of simultaneous threads to perform multiple tensor operations rapidly. This is in contrast to the central processing unit (CPU), which typically contains only a handful of cores. GPUs are also optimised for much higher memory bandwidth than CPUs. Over the past few years, specialised applications which are both memory intensive and highly parallelisable, outside of graphical rendering, have also received significant performance boosts when executed on a GPU. This type of operation is referred to as general purpose GPU computing (GPGPU). These applications include weather simulations, computational chemistry and lately deep learning. As shown in Equation 3.48 the propagation of information through a dense feed forward neural network can be broken down into linear transformations and tensor operations. Therefore, the training of deep neural networks sees massive performance boosts using hardware acceleration.

3.7.3 Data Scaling

In deep learning, the inputs of a model may have different units and thus could have vastly varying scales. The network's trained in Chapter 9 include inputs such as the number of reconstructed leptons, as well as several p_{T} measurements. The former variable has values which range from zero to five, while the latter variable measured in MeV ranges from zero to several hundred thousand. This discrepancy may hinder the training process for a number of reasons, as the distributions of the input features affect the activations throughout the network.

The input neurons, and those of the first few layers, may become over-saturated if the scales of the features are too large. Inputting values of a few thousand to an activation like Sigmoid or tanh will produce very small gradients, resulting in a very slow learning network. An over-saturated neuron with an activation like RelU, ELU, or Swish becomes linear, limiting the capacity of the network.

Furthermore, if the different features have drastically different scales, then the one with the largest range will have the greatest effect on the output of the network. In the example above, it would be numerically favourable to modify weights associated with the p_{T} measurements. This could cause it to dominate the training process, while other inputs, such as the lepton multiplicities, are relatively ignored.

To combat this effect, a pre-processing step called data scaling can be applied. This step reduces the input values to similar ranges which centre on zero. Data scaling is performed using different values for each feature. These values are often the min, max, mean and variance derived from the training set. These values are then saved along with the network and any new data must also go through the same pre-processing step. This also removes the dimensions of the inputs making them all unitless.

The two most commonly used methods are:

1. **Min-Max Normalisation.** Each input is individually scaled to fall within the range of $[-1, 1]$, meaning that the original distribution shape is preserved. This method uses the mean and maximum values of each input distribution and is thus very sensitive to outliers.

$$x_{\text{new}} = \frac{x - \min}{\max - \min} \tag{3.49}$$

2. **Standardisation.** Each input is individually scaled so that the new distributions have zero mean and unit variance. This does not guarantee a common numerical range, but it does reduce the effects of differing scales.

$$x_{\text{new}} = \frac{x - \mu}{\sigma} \tag{3.50}$$

In addition to scaling the input features of neural networks, it is also common to scale the output features of $\mathbf{y}$ and $\hat{\mathbf{y}}$. This ensures that each output feature contributes to the loss with relatively similar scales. Otherwise the network would just prioritise the output feature with the largest scale.

3.7.4 Parameter Initialisation

As previously stated, the initial values of the trainable parameters of the network θ are random. This starting point can determine the convergence rate of the algorithm, if it converges to a local minimum with high or low cost, or whether it converges at all. A proper initialisation scheme is crucial for an effective training process.

Various heuristics are available for choosing the initial scale of the weights, most of which strive to ensure that the inputs to each layer of neurons have a mean of zero and around unit variance. Most commonly used is the heuristic where the weights between layer l and $(l+1)$ are drawn from the uniform distribution:

$$U\left(-\frac{1}{\sqrt{l}}, \frac{1}{\sqrt{l}}\right). \tag{3.51}$$

The biases of the network are similarly initialised. This scheme ensures that if activations of the previous layer have zero mean and unit variance, the inputs to the next will be similarly distributed.

Many other initialisation schemes exist and in some cases are strongly advised. The original paper for the SELU activation function [111] noted that using a normal distribution $\mathrm{N}(0, 1/\sqrt{l})$ instead of a uniform distribution for the weights and initialising all bias values with zero produced better results.

3.7.5 Invariances

One of the main concerns with deep learning is how the model will deal with invariances in the data [134]. This is especially true for problems involving pattern recognition or physical systems with inbuilt symmetries. An example may exist where the predictions of a model should be invariant under some transformation of its features. These transformations would change the input values given to the network and thus change the initial activations of the neurons, and yet the result must remain the same. For image recognition models, a particular object should be classified the same way irrespective of its position or orientation in the image. The invariance encountered in Chapter 9 is the nominal cylindrical symmetry of the ATLAS detector. Whereby the magnitude of the measured $E_{\mathrm{T}}^{\mathrm{miss}}$ should be invariant under a rotation of the event about the beam axis.

Certainly, if sufficiently large numbers of training samples are available, then an adaptive model such as a neural network could learn the invariance on its own, or at least learn how to approximate it. Very little domain knowledge would need to be manually provided, but the training set must include many examples of the effects of the transformations. However, this approach may be impractical due to the limited number of training samples or if there exist many invariants which combine exponentially. There are several approaches that could be used to encourage an adaptive model to learn the invariances and the underlying structure of data. These approaches can be divided into three categories [74].

1. The model is left to learn the patterns and structure of the data on its own, but this is helped by augmenting the dataset. Here the training set is expanded to include transformed states of the system. This augmentation technique is randomly reapplied between each training iteration. For example, in image recognition, each image in the training set is randomly rotated, flipped and cropped between each epoch.

2. The invariance is built into the structure of the neural network. This is one of the main motivations for convolutional neural networks which use local receptive fields. The same feature extraction is performed across the whole image, exploiting the translational invariance of objects in images.

3. Data is pre-processed and only features which are invariant under the required transformations are provided to the model. Any subsequent system would therefore respect the invariances.

The benefit of each approach is dependent on the type of problem, the complexity and range of transformations, and the availability of invariant features. However, studies have shown that constructing a basis of invariant inputs (option 3) can yield higher performance at significantly reduced computational costs compared to augmenting the data (option 1) [134].

3.7.6 Batch Normalisation

Batch normalisation (BatchNorm) is a recently developed method for the adaptive reparameterisation of network activations and has been met with widespread success. But exactly how and why BatchNorm helps in training deep networks is unclear [135–139]

Batch normalisation was initially proposed to combat a phenomenon called internal covariate shift (ICS) [139]. Gradient descent is performed on all parameters of the neural network based on a single measurement of the cost function. Each parameter is updated under the assumption that it is the only one changing, and this can lead to some unexpected results. One could in principle recalculate the cost and gradient after every individual parameter update, but this would be unfeasible for larger networks. In practice, within the same iterative step, the distribution of the inputs given to later layers change due to parameter updates in earlier layers. Therefore, the descent step applied to each later layer is no longer optimal. These fluctuations in the distributions is referred to as ICS and it makes neural networks much harder to train.

BatchNorm can be applied to any hidden layer and was claimed to reduce the problem of ICS by coordinating updates across the network. A BatchNorm layer performs a normalisation process for each mini-batch during forward propagation. The explicit steps of a BatchNorm layer, which is applied after the activation function, is shown in Equation 3.53. It uses the mean μ_b and variance σ_b^2 of each activation across a mini-batch. It then normalises the activations of each neuron a_i and reparameterises them using trainable parameters γ and β to produce its output $\widetilde{a_i}$. Here a_i represents the activation of the same neuron for training example i within the minibatch.

$$\text{1) Calculate activation mean:} \qquad \mu_b = \frac{1}{M}\sum_{i=1}^{M} a_i \tag{3.52}$$

$$\text{2) Calculate activation variance:} \qquad \sigma_b^2 = \frac{1}{M}\sum_{i=1}^{M}(a_i - \mu_b)^2 \tag{3.53}$$

$$\text{3) Reparameterise:} \qquad \widetilde{a_i} = \gamma\frac{a_i - \mu_b}{\sqrt{\sigma_b^2 + \epsilon}} + \beta \tag{3.54}$$

$$\tag{3.55}$$

Here ϵ is a small constant to prevent the division of zero. During training time, the means and standard deviations are calculated per batch. The gradients of the cost function are propagated through these steps. After training, the final model replaces μ_b and σ_b^2 with running averages, allowing the reparameterisation to take place on batches of single examples.

The creators of BatchNorm claim that fixing the distribution of the activations between mini-batches reduces the effects of ICS. While there is still debate on how exactly batch normalisation helps with learning, there are now strong arguments that it has very little to do with ICS [136, 138]. The parameters γ and β define the mean and variance of the new activation distributions, and since they are trainable parameters, these distributions will vary with each iteration, directly causing ICS. However, since only these two values define the statistics of an entire layer, the underlying optimisation problem becomes more stable [138].

Batch normalisation has been demonstrated to significantly increase the speed of convergence, since the gradients are more constrained and well behaved. Furthermore, it was shown that the landscape of the optimisation problem became significantly smoother in models with BatchNorm, ensuring that the gradients were more predictive. This allows for significantly higher learning rates without the risk of divergence [135]. Batch normalisation also allows for arbitrary weight initialisation. Additionally, it has been shown that batch normalisation improves generalisation and acts as a regularisor with similar effects to dropout [135, 137].

3.7.7 Hyperparameter Optimisation

The hyperparameters of a model are those whose values were set by the user before training. They are distinct from trainable parameters since they are not adapted and modified by the learning algorithm itself. They do however have a massive impact on the algorithm's behaviour and thus need to be optimised. Hyperparameters can be broadly split into two categories. First are those which describe the network architecture and structure. These include the model depth, layer widths, the inclusion of BatchNorm or dropout layers, or type of activation functions found throughout the network. Second are those which are specific to the training procedure, such as the learning rate η, the mini-batch size M, the type of loss function, the parameter norm penalty parameter α, or the dropout probability p.

Searching for the best combination of hyperparameters is an exhaustive process, as there is no strong theory on what the optimal choice might look like. It is highly particular to the task at hand. Even the best form of optimiser is still under debate. While adaptive learning rate algorithms, particularly Adam and RMSProp, have become the standard for deep learning, there have been studies showing that these techniques fail to find optimal solutions and are outperformed by SGD with momentum given enough training time [140]. Most notably, when the number of parameters is of the same order as the number of training examples, the use of an adaptive optimiser can result in a model with a poor ability to generalise.

Hyperparameter searches require an additional collection of examples other than the training and the evaluation set. This is usually the same testing set used for early

stopping as described in Section 3.6.1. The search involves trying a new combination of hyperparameters, fully training the model, and then evaluating its performance on the testing set. Since this requires a completely fresh model to be trained from scratch for each combination of hyperparameters, it is an immensely time-consuming process. There are several techniques to select the next combination of hyperparameters to test during the search.

The traditional method is called a grid search. This is a full parameter sweep of a manually selected subset of the hyperparameter space. Grid searches suffer from the curse of dimensionality, but they are also embarrassingly parallel. However, the manually selected subset requires a discretisation of the hyperparameter space, so not all possible combinations can be tested.

An alternative approach is to use Bayesian optimisation. Here a second statistical model is built which tries to approximate the mapping from the set of hyperparameter values to the final testing error. By iteratively building a history of hyperparameter configurations, Bayesian optimisation selects the combination that shows the most promise. The process however is slow, as it must sequential, and it can only handle numeric parameters.

Another method is the random search. Here, networks are trained using random values of the hyperparameter space. This process can be parallelised like the grid search, but it can also be performed on a continuous hyperparameter space.

The final hyperparameter optimisation strategy is the most widely adopted by researchers, students and hobbyists. It is sometimes called babysitting or grad-student descent (GSD). Here an individual tweaks subsequent networks in the small hope that improvements are made. This approach is completely manual and while it is easy to implement, it is often only slightly better than a random search. There has also been no proof of convergence, so the process continues until one runs out of time or motivation.

Chapter 4

CERN and The Large Hadron Collider

The European Organisation for Nuclear Research, most commonly referred to as CERN[†], was founded in 1954 as a partnership between twelve countries [141]. Today CERN houses an entire accelerator complex which is shown in Figure 4.1. Since its founding over sixty years ago, the organisation has grown to contain 22 member states, 8 associate members and 6 observer groups. There are over 15 000 staff, fellows, associates, and users, making it one of the largest purely scientific and research based organisations in the world.

CERN's greatest achievements predominantly lie in the realm of physics, such as the discovery of the W boson [142], Z boson [143] and Higgs boson [4, 36, 37], but the organisation has also overseen some considerable advances in many other fields, especially computer science. CERN produces massive amounts of data which need to be stored, analysed and provided to its members around the globe. In 2018 alone CERN generated 72 petabytes of data [144]. To cope with this monumental task, CERN designed The LHC Worldwide Computing Grid, the largest of its kind, comprising of over 170 computing facilities across 42 different countries [145]. The World Wide Web, a tool used by billions of people across the globe, started at CERN in 1990 [146]. CERN's achievements not only propelled fundamental physics and pushed the frontiers of technology, they have also played a key role in the development of the Information Age.

The LHC [148] is the world's largest and most powerful particle collider. Its purpose is to examine the smallest building blocks of matter and the forces which exist between them by colliding protons, and occasionally lead-nuclei, at almost the speed of light. The machine was designed to be able to produce pp collisions at a centre-of-mass of 14 TeV. Its collisions have set new heights on man-made temperatures [149, 150] and the device itself is the largest machine mankind has built to date [151]. It is a circular collider built underground, with a depth ranging from 45 to 170 meters and

[†]From the original French name of the council setup in 1952 to oversee the construction of the new laboratory: the Conseil Europeén pour la Recherche Nucléaire. The name was officially changed two years later at the organisation's founding, but the acronym was kept.

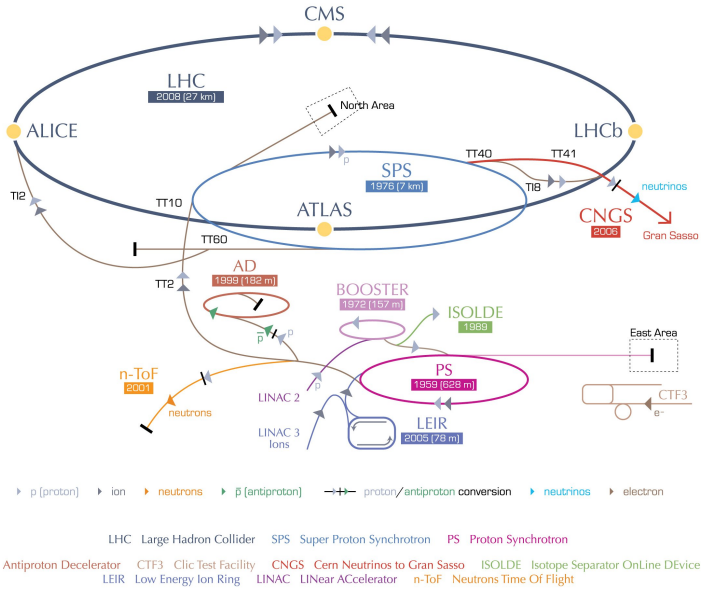


Figure 4.1: A diagram illustrating the layout and contents of CERN's accelerator complex [147].

a circumference of 26.7 kilometres [152]. It was completed in 2008 after ten years of construction, making it the most recent collider added to the CERN accelerator complex. The LHC is the second accelerator to be housed in the tunnel underneath France and Switzerland, the first being the Large Electron Positron (LEP) collider which was dismantled in 2000.

The collider tunnel contains two adjacent parallel beam pipes segmented in 8 octants, shown in Figure 4.2. Each of these pipes carries a proton beam which travel in opposite directions around the ring. The beams are intersected at four different interaction points, causing high energy collisions. Encompassing each interaction point is one of the four main LHC particle detectors. These four main experiments and their respective locations along the LHC ring can be seen in Figures 4.1 and 4.2. ATLAS [153] and CMS [154] are two all-purpose detectors, and are therefore used to study a wide variety of physics processes. Most notably, both discovered the Higgs boson in 2012 [4, 36], the last particle required to complete the Standard Model. Current work at these experiments involves refining measurements within the Standard Model and looking for evidence of BSM, such as SUSY or dark matter. The two detectors have complementary designs, which allow cross-validation between the two experiments. The ALICE [155] and LHCb [156] experiments have detectors designed to probe a specific branch of physics. LHCb has an asymmetrical detector which is specifically designed to detect B Hadrons, and therefore its focus is on measuring CP violation and the asymmetry between matter and anti-matter [156]. The ALICE detector is designed to study heavy ion collisions. It is through these heavy ion collisions that ALICE hopes to gain a deeper understanding of the Quark Gluon Plasma (QGP), a state of matter consisting of asymptotically free strong-interacting quarks

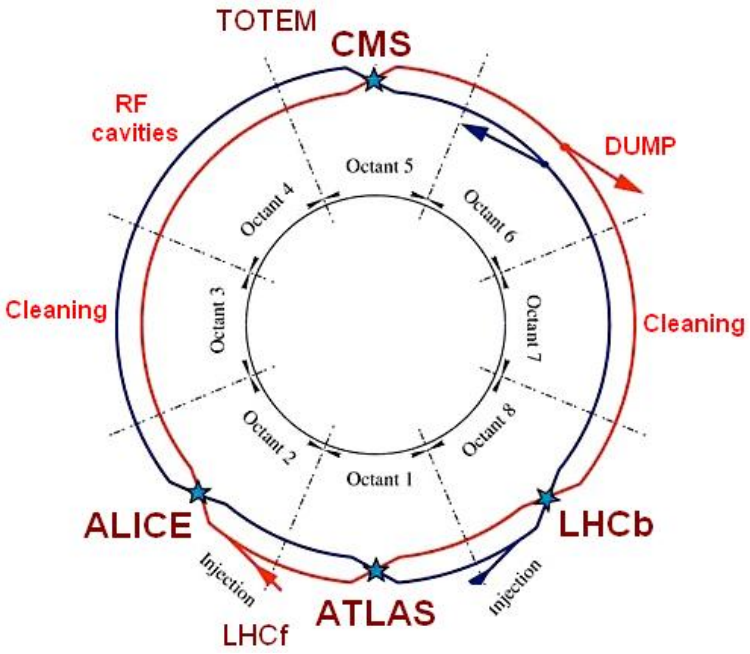


Figure 4.2: A diagram showing the LHC, the locations of the detectors, and the nature of its structure divided into octants [157].

and gluons [155].

In addition to these main four, there are three smaller and complementary experiments at the LHC which share an interaction point with one of the larger ones. Located around 140 meters either side of ATLAS, the detectors of the LHCf experiment [158] observe particles scattered extremely close to the beam pipe. It can use these measurements to compare the shower models used to estimate the energy of cosmic rays. The TOTEM [159] experiment shares an interaction with CMS, and it aims to precisely measure the pp cross-section. Finally, the seventh and newest LHC experiment, MoEDAL [160] is dedicated to searching for highly ionising particles belonging to BSM physics and magnetic monopoles. It was built alongside LHCb and started recording data in 2015.

To its main experiments, the LHC provides pp collisions by crossing the oppositely moving beams so that that pairs of bunches move through one another. Each bunch has on average 115 billion protons and the bunch-crossing takes 4 ns. These bunches are collided at a rate of 40 MHz [148].

Luminosity

The following section covers the concept of luminosity as presented in References [161–163]. While the energy is the most important parameter in collision experiments [161], the quantity of useful interactions produced in a collider is dependent on its luminosity. One of the greatest achievements of the LHC is that it can provide pp collisions at both unprecedentedly high centre-of-mass-energies and luminosities [164]. Luminosity is either quoted as instantaneous or as integrated as both are used

to assess the performance of a collider. Given a particular interaction of interest with cross-section σ, and collider with instantaneous luminosity $\mathscr{L}$, the rate at which the interaction takes place, dN/dt, can be simply calculated by

$$\frac{dN}{dt} = \mathscr{L}\sigma\,. \tag{4.1}$$

Therefore, the unit of luminosity is usually $\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$.

If the colliding beams have identical energies and Gaussian shapes, as is the case in the LHC, the instantaneous luminosity can be expressed as,

$$\mathscr{L} = \frac{N_p^2 N_b^2 f}{4\pi\sigma_x\sigma_y} S. \tag{4.2}$$

Here N_p is the number of protons in each bunch, N_b is the number of bunches, f is the bunch-crossing frequency, and σ_x and σ_y are the beam widths in the transverse plane [161]. The final term S is a luminosity reduction factor brought into play when considering that the beams at the LHC are not colliding perfectly head on. Rather, the bunches cross at a finite angle α. This angle is necessary as it allows for the separation of the bunches when they are away from the interaction point while still sharing the same beam pipe. However, the downside to this is that the experiments like ATLAS and CMS only experience around 65% of the luminosity achievable with truly head on collisions [165]. The total number of interactions N can be derived using the total integrated luminosity,

$$N = \sigma\mathscr{L}_{int} = \sigma\int \mathscr{L}(t)dt. \tag{4.3}$$

In 2011 the performance goals of the LHC stated that they planned to be able to deliver a peak instantaneous luminosity of $10^{34}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ to the ATLAS and CMS experiments [152]. Towards the end of 2018, CERN reported that in almost every run since late 2017 it achieved more than double this [166]. The total integrated luminosity since the beginning of LHC operation up to October 2018 was at $189.3\,\mathrm{fb}^{-1}$, of which $160\,\mathrm{fb}^{-1}$ were accumulated during Run 2 [167].

Chapter 5

The ATLAS Experiment

5.1 Overview

The ATLAS[†] experiment is one of the main four experiments currently in operation at the LHC. The following section describes the ATLAS detector in its current form, covering its components and their functions. The information presented is a summary of the ATLAS Technical Design Report [153].

The ATLAS detector is an all-purpose particle detector which lies at Point 1 of the LHC ring. It was designed to identify almost all of the final state particles produced during the high energy pp collisions supplied by the LHC and record their kinematics. The structure is the largest experiment at CERN, measuring 44 metres long, 25 metres in diameter and weighing over 7000 tonnes. By volume it is the largest particle detector ever constructed for collision experiments.

The ATLAS detector contains multiple sub-detectors arranged in a series of concentric cylinders surrounding the interaction point - the location where the proton beams of the LHC collide. It also includes end-cap components for each sub-detector to better cover the forward regions of the collisions, capturing particles ejected closer to the proton beam. Each of these sub-detectors are made up of many layers and are specially engineered to detect different particles, or have specific roles in the reconstruction process. The entire detector is nominally forwards-backwards symmetric along the beam pipe. A cutaway view with the various sub-detector levels visible is shown in Figure 5.1.

The main sections of ATLAS are the Inner Detector (ID) [168, 169], the calorimeters [170], the Muon Spectrometer (MS) [171] and the magnet system, which is comprised of a central solenoid [172], a barrel toroid [173] and two end-cap toroids [174].

All of these detector subsystems are complimentary. The ID can track the motion of charged particles accurately, the calorimeters are able to stop and measure the energy of most objects coming out of the collision, and the MS takes additional readings of highly penetrating muons. ATLAS has two magnet systems. The first is

[†]The name ATLAS was originally the acronym: A Toroidal LHC ApparatuS. However, this has since been dropped and "ATLAS" is now simply the name of the experiment.

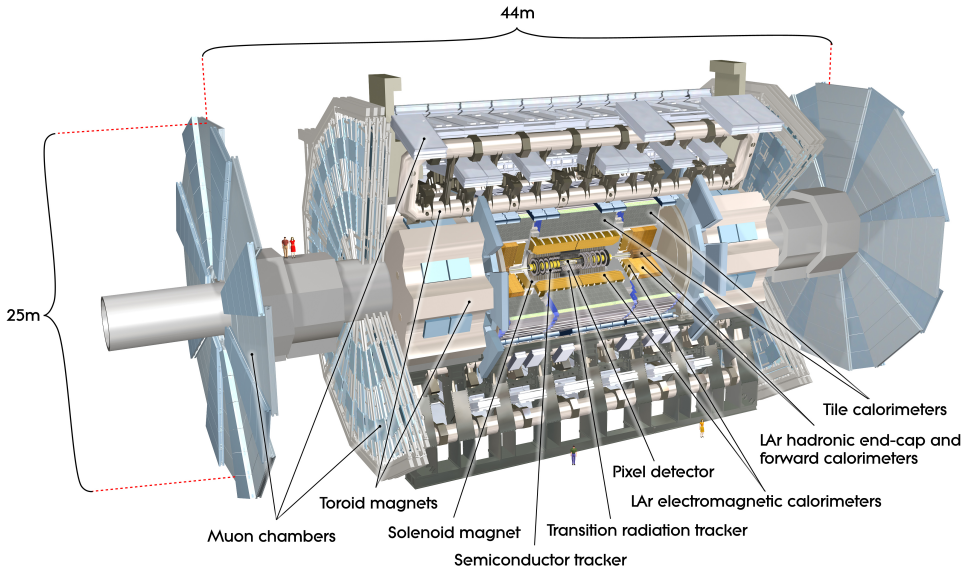


Figure 5.1: A cutaway diagram of the ATLAS detector [153].

a superconducting solenoid which bends charged particles moving through the ID and the second is a system of three toroidal magnets which bend muons travelling through the MS. Due to the Lorentz force, a singularly charged particle ($|q| = e$) moving through a uniform magnetic field B will trace out a helix with radius R and pitch angle λ, related to its momentum by Equation 5.1. Thus, the momentum p of the particle may be derived. The design of ATLAS was based around these magnet systems.

$$p \cos \lambda = 0.3BR \tag{5.1}$$

The combination of all of these sub-detectors means that the only established particles in the Standard Model which may not be directly detected are neutrinos. This is due to their small interaction cross-sections with hadronic matter. However, their presence may be inferred through the momentum imbalance among all of the observed particles, as discussed in Chapter 7. This method may only work if all final state particles produced in a collision are detected and correctly identified. Hence the importance of the end-cap regions of the detector so ATLAS can cover almost the full 4π solid angle of a collision. This makes it an example of a hermetic detector, in that all non-neutrinos produced in the collision are reconstructed, with very few blind-spots. Therefore, it is a serious requirement that all detector subsystems must be maintained for reliable data to be taken. This is an engineering challenge considering the high radiation areas immediately surrounding the proton collisions.

5.1.1 Coordinate System

The following brief section describes the coordinate system currently used by the ATLAS experiment. This is necessary since the nomenclature is used throughout the

rest of the document. The origin of this coordinate system is taken to be the nominal interaction point. The x-axis points from this origin towards the centre of the LHC ring and the y-axis points vertically upwards. The z-axis is parallel with the LHC beam pipe running through the centre of the detector. The positive z direction is taken so that the coordinate system is right-handed, and therefore it is in the anti-clockwise direction of the beam pipe. The side of the ATLAS detector which can be defined with a positive z value is referred to as side-A, with the opposite being side-C. Since the incoming beams travel parallel to the z-axis, the $x-y$ plane defines the transverse plane of the interaction. The use of cylindrical coordinates is also employed. The azimuthal angle ϕ is measure around the beam pipe and therefore can be described by $\phi = \tan^{-1}(y/x)$. The value $r = \sqrt{x^2+y^2}$ is also used to describe the transverse distance from the beam axis.

In spherical co-ordinates the polar angle θ is the angle from the z-axis. An additional and commonly used spacial coordinate is the pseudorapidity η, which is an approximation for rapidity y_r in the relativistic limit. Rapidity is defined as

$$y_r = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}\,. \tag{5.2}$$

When the mass term becomes negligible compared to its momentum, as is the case for many of the particles coming off the high energy collisions at the LHC, then the equation becomes the definition for pseudorapidity

$$\eta = \frac{1}{2}\ln\frac{|\mathbf{p}|+p_z}{|\mathbf{p}|-p_z} = -\ln\frac{\theta}{2}\,. \tag{5.3}$$

In hadronic colliders the partons which produce the underlying collision may carry different longitudinal momentum fractions. This implies that the rest frame of the parton-parton collision may have different longitudinal boosts. Therefore, describing a coordinate system using pseudorapidity is more convenient than using θ, as differences in rapidity are Lorentz invariant. Furthermore, particle production in collider experiments are relatively constant as a function of rapidity. Angular separation between points in this coordinate space are defined by $\Delta R = \sqrt{\Delta\eta^2+\Delta\phi^2}$. This value is also Lorentz invariant in the massless limit.

5.2 Inner Detector

The ID is the closest sub-detector to the beam pipe, and therefore is the first part of ATLAS that final state particles emerging from the collision will encounter. It has a radius of 1.05 m and a length of 6.2 m. Like the rest of the ATLAS detector it is centred on the interaction point and it is symmetrical under reflections along the beam axis. A schematic of the ID is shown in Figure 5.2. The ID is crucial for particle identification, momentum measurements and vertex construction. Permeating the entire sub-detector is an axial magnetic field of 2 T, provided by the ATLAS central

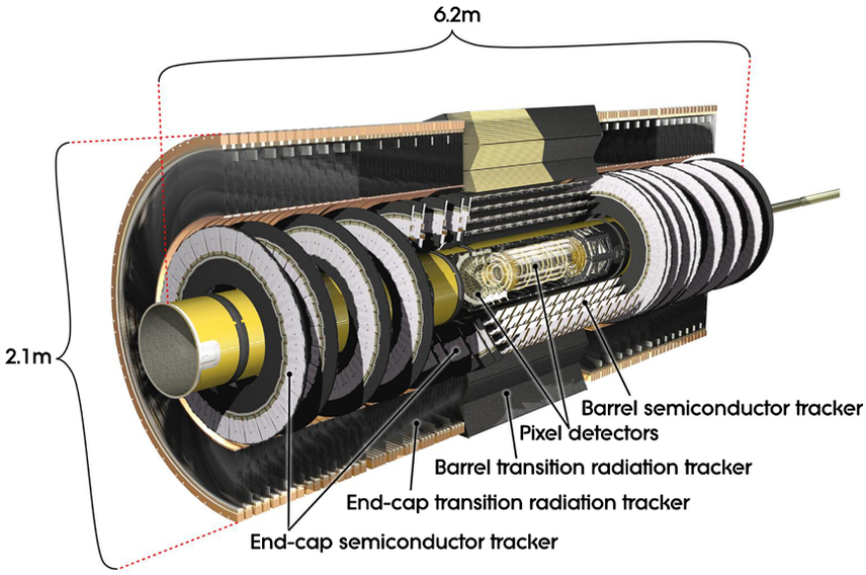


Figure 5.2: A cutaway view of the Inner Detector such that all three major components are shown; the Pixel Detector, the Semi-Conductor Tracker and the Transition Radiation Tracker [183].

solenoid. The function of the ID is to track charged particles by mapping out their discrete interactions with many of its layers in 3D space. The multiple interactions or hits are used to reconstruct the track the charged particle traced out as it travelled away from the interaction point. This track should be helical due to the axial magnetic field. Hence, the particle's momentum and the sign of its charge may be derived. Since most of the ID components require the incoming particles to generate ionising radiation to produce a detector signal, only charged particles can be measured using the ID.

The ID itself is comprised of three independent yet complementary components: the Pixel Detector [175, 176], the Semi-Conductor Tracker (SCT) [177–179] and the Transition Radiation Tracker (TRT) [180–182].

5.2.1 Pixel Detector

The Pixel Detector is the innermost component of the ID. Due to its close proximity to the interaction point, it requires the highest granularity. The function of the Pixel Detector is the reconstruction of vertices through track extension. A vertex is where extrapolated tracks intersect close to the interaction point. This indicates a point in space where more than one detected particle emerged and reveals the location of an underlying interaction. The reconstruction of primary vertices can help distinguish between particles emerging from pileup interactions as explained in Section 6. The reconstruction of secondary vertices is crucial for b-tagging jets.

The pixels are made from oxygenated n-type silicon, each has a surface area of 50 µm $\times$ 400 µm, and a width of 250 µm. The small pixel sizes allow for highly precise track resolution. As a highly energetic charged particle travels through a pixel, it ionises

the silicon leaving free electrons and positively charged holes in the semi-conductor material. An applied voltage causes these charges to separate and move the edges of the pixel. There they are received by electronics which read out a small current, registering that the pixel was hit.

Batches of 47 232 pixels are grouped into a single sensor chip, of which there are 1744 in the detector. This results in over 82 million readout channels in the Pixel Detector, which is over half of all the readout channels in ATLAS. The sensors are arranged in three concentric cylindrical layers in the barrel region of the detector. There are a further three disks of sensors on both end-caps. The Pixel detector records only around three measurements for a single charged particle, yet it is able to provide a position resolution of 10 µm in the $r - \phi$ plane and 115 µm along the $z-$axis.

Another consequence of the detector's proximity to the proton-proton collisions is that the electronics are exposed to high doses of radiation. During construction, these components had to be radiation hardened. Despite these efforts, the innermost layer of the ID, called the B-Layer, was designed to be replaced every three years due to radiation damage. In 2015 a new component of the ID was installed. The Insertable B-Layer (IBL) [184] was placed even closer to the interaction point, adding a fourth layer to the Pixel Detector. To make room for this new component, the beam pipe within ATLAS was decreased in diameter by 6 mm. The IBL was added to the pixel detector to improve tracking performance and to prepare it for the higher luminosities that would be provided during Run 2. Its insertion also allowed the previous B-Layer to remain operational instead of having to be replaced.

5.2.2 Semi-Conductor Tracker

The Semi-Conductor Tracker (SCT) is the middle component of the ID. It performs largely the same function and utilises the same technology as the Pixel Detector it encompasses. However, the SCT makes use of bigger silicon strips instead of pixels to cover a greater area. Each module of the SCT has a double layer of sensors, whose axes are perturbed by about 40 mrad from each other. Using the combined pair of measurements, each layer provides a longitudinal resolution of 580 µm and a transverse resolution of 17 µm. There are 15 912 sensors arranged in four layers in the barrel region and nine disks in each of the end-cap regions, totalling over 6.3 million readout channels. A single particle ejected at $|\eta| < 2.5$ may provide between four to nine measurements in the SCT.

5.2.3 Transition Radiation Tracker

The Transition Radiation Tracker (TRT) is housed in the outermost layer of the ID and is made up of a collection of straw tube detectors, as shown in Figure 5.3. Each of these straw tubes are 4 mm in diameter and contain a coaxial gold-coated tungsten wire running down its length. The TRT is comprised, like the other components, of

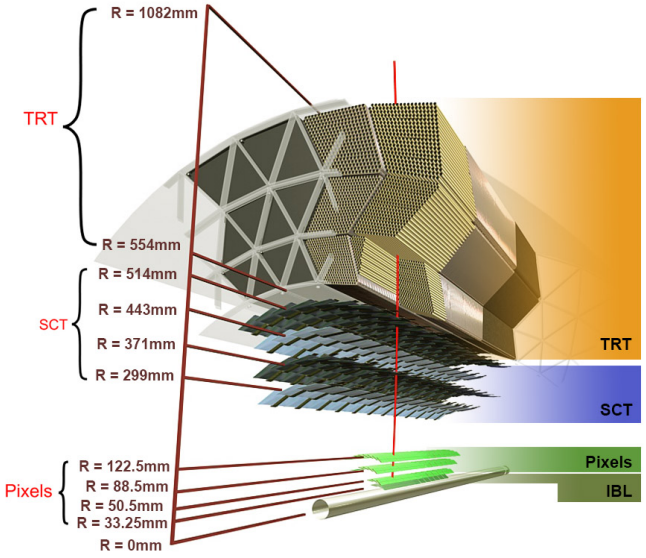


Figure 5.3: A schematic of the barrel region layers of the Inner Detector, including the new Insertable B-Layer [185]. The drawing includes the track, in red, of a particle with $p_T = 10\,\text{GeV}$. The particle leaves the interaction point, and traverses the IBL layer, three pixel layers, four double sided SCT layers, and around 35 TRT barrel straws.

a barrel and end-cap segments. In the barrel region, the tubes are arranged parallel to the beam pipe and are split into two groups at $z = 0$. In the end-cap sections the tubes are arranged radially. The outer shell of each straw acts as a cathode while the tungsten wires in each centre act as anodes, with a 1500 V potential difference between the two. The straws are filled with a gaseous mixture which is 70% Xe, 27% CO_2, and 3% O_2. As charged particles cross the tubes, they ionise the gas within. The free electrons then drift towards the wires in the centre, creating an electrical pulse which indicates that the straw was hit.

The TRT is not only a charged particle tracker, but also serves as an electron identification detector [186]. Between the straws there are materials with widely varying indices of refraction, which cause fast-moving charges to emit transition radiation. This radiation, in the form of X-rays, interacts with the Xenon gas mixture and provides much stronger signals than if ionising radiation was the only process at play. The strength of this extra transition radiation signal is inversely proportional to the mass of the particle, thus the TRT can distinguish between light electrons and heavier composite particles such as charged pions.

The position resolution of the TRT is not as good as the Pixel Detector nor the SCT, but the technology was used to reduce the cost of filling a larger volume in the detector whilst maintaining tracking capabilities. While the barrel component of the TRT provides transverse position measurements to an accuracy of 130 mm, it is not capable of providing information in the longitudinal direction. In the same way, the TRT end-cap regions can only provide information about the particle's position in the $z - \phi$ plane. Despite this, the TRT contributes significantly to the momentum measurement of the detected particles as it contains over 351000 readout channels

and on average 36 will be activated for a single track.

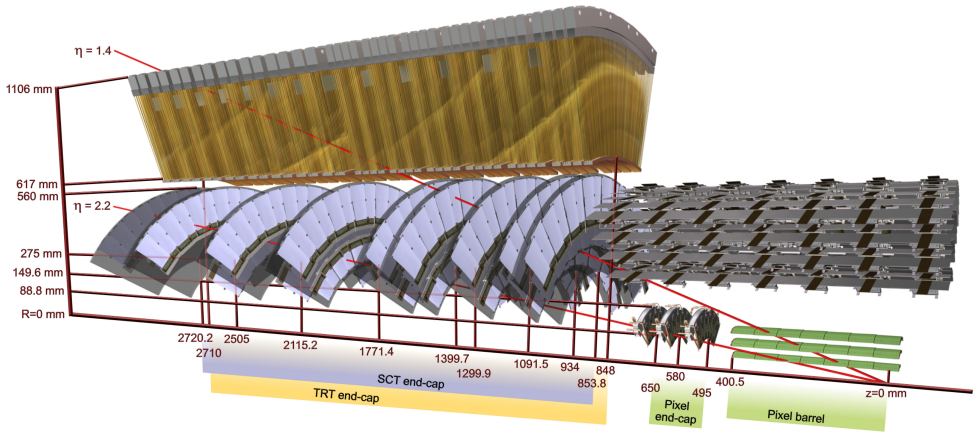


Figure 5.4: An illustration of the ID elements, focusing on one end-cap region [187]. It also shows the components encountered by two charged particles with $p_T = 10\,\text{GeV}$. Their tracks are shown in red. The particle emitted at $|\eta| = 1.4$ comes into contact with three pixel layers, four end-cap SCT disks, and several straws in the TRT end-cap. Another particle, at $|\eta| = 2.2$, only interacts with one layer in the Pixel Detector, and the last four disks of the SCT end-cap. It is important to note that this schematic does not include the IBL which was installed in 2015.

5.3 Calorimeters

The ATLAS calorimeters sit between the ID and the muon spectrometer, just outside the solenoid magnet. Their function is to provide particle identification and energy measurements for photons, electrons, jets and other hadrons. They are therefore crucial in E_T^{miss} reconstruction. The various components of the calorimeter system at ATLAS have a combined coverage of $|\eta| < 4.9$. The system is made up of three independent calorimeters, each with several components, all shown in Figure 5.5. The Electromagnetic Calorimeter (ECal) is comprised of a single barrel region and two electromagnetic end-cap regions. The Hadronic Calorimeter (HCal) contains a tile barrel section, two tile extended barrel sections, and two hadronic end-cap (HEC) regions. Finally, there is the Forward Calorimeter (FCal). These calorimeters employ the same sampling principle to measure the energies of the particles they capture. They contain alternating layers of high-density metal, which absorb energy from incoming particles to create a particle shower. In between these absorbing layers lie active materials which can record the shape of the shower and the amount of deposited energy. Only a fraction of the energy produced by the particle is "sampled" by the active sensor, but the full shower energy may still be inferred. The active material used in the ECal, the FCal and the HEC sections is liquid argon (LAr). This material is chosen due to its intrinsic radiation hardened properties as well as its linear response.

The ECal is located closest to the interaction point and measures the energies of lighter particles which interact via the electromagnetic force. Its LAr scintillators are

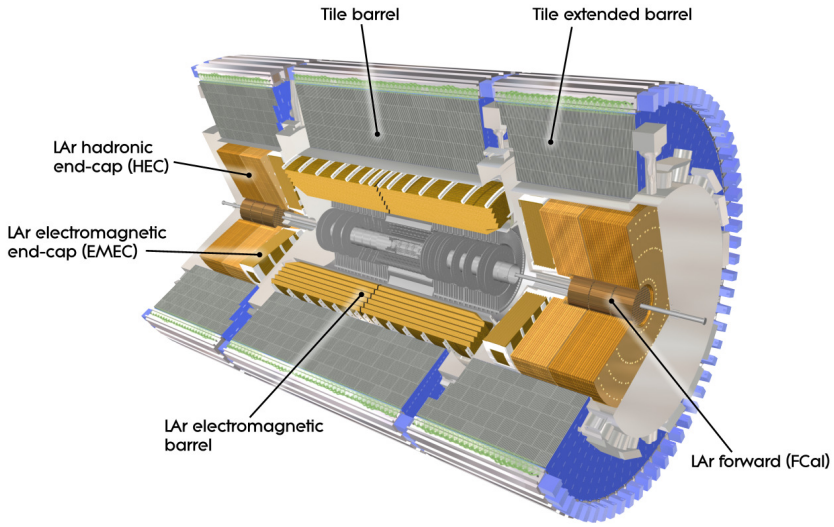


Figure 5.5: A cutaway diagram of the ATLAS calorimeter system [170].

interleaved with dense lead plates. These lead plates encourage particle shower formation as they cause high energy photons to pair-produce and fast-moving charged particles to emit bremsstrahlung radiation. These layers are arranged in an accordion shaped geometry, which ensures full ϕ coverage of the detector, enables fast signal extraction, and provides information about the shower's longitudinal evolution. The ECal performs with high precision, both in energy estimates and energy localisation, with an energy resolution of $10\%/\sqrt{E} \oplus 0.2\%$ and an angular resolution in the η plane of 50 mrad.

The HCal measures the energies of particles which are able to pass through the ECal yet interact via the strong nuclear force. Due to the much larger distance between nuclear interactions, hadronic showers occupy greater volumes than electromagnetic showers, and the HCal has to accommodate for that space. Many of the features of the HCal were chosen for their cost-effectiveness, due simply to the amount of material which was required for its construction. The HCal is also designed to prevent any particle or any shower from reaching the muon spectrometer.

The largest component of the HCal is the tile barrel section, which is 8 m in diameter and encompasses 12 m of beam axis, providing a coverage of $|\eta| < 1.0$. This coverage increases to $|\eta| < 1.7$ with the inclusion of the tile extended barrel region. Both the tile barrel calorimeter and the tile extended barrel calorimeter use scintillator tiles and steel plates for the active and absorber materials respectively. The energy resolution for the tile calorimeters is $50\%/\sqrt{E} \oplus 3\%$. The HEC calorimeter provides coverage from $1.5 < |\eta| < 3.2$ and uses a copper absorber. The energy resolution for the HEC differs depending on the type of particle.

Finally, the FCal uses a copper and tungsten absorbing material. It is located in the far forward regions, close to the beam pipe, providing energy measurements

of particles ejected with η as high as 4.9. The energy resolution for the FCal is $21\%/\sqrt{E} \oplus 3.5\%$ for electrons and $70\%/\sqrt{E} \oplus 3.0\%$ for pions. It provides both electromagnetic and hadronic energy measurements.

5.4 Muon Spectrometer

The MS is contained in the outermost layer of the ATLAS detector, surrounding the calorimeters. Its function is to measure the position of muons, which are able to traverse the calorimeters largely unimpeded since, due to their large mass, they do not emit much bremsstrahlung radiation. ATLAS was designed to contain all stable particles other than muons and neutrinos within the calorimeters. Since neutrinos are highly unlikely to interact with the hadronic matter of the ATLAS detector, only muons should leave any trace in the MS.

The MS resides within a $0.5 - 1.0\,\mathrm{T}$ magnetic field created by three toroidal magnets. The barrel toroid covers the range $|\eta| < 1.6$, and exhibits an eight-fold rotational symmetry, while the two end-cap magnets are located in the range $1.6 < |\eta| < 2.7$. Tracks of muons are bent by both magnets in the transition region, $1.4 < |\eta| < 1.6$.

In addition to the magnet system, there are four distinct detector components that are used in the MS. These include Monitored Drift Tubes (MDTs) [188] and Cathode Strip Chambers (CSCs) [189], both of which are used for highly precise position measurements and tracking. Resistive-Plate Chambers (RPCs) [190] and Thin-Gap Chambers (TGCs) [191] are used for dedicated triggering based on momentum measurements and for making immediate decisions about whether the data captured during the bunch-crossing is worth saving. All four of these components, as well as the three toroidal magnets are shown in Figure 5.6. The muon spectrometer has cylindrical barrel layers of detectors. Figure 5.7 also shows three end-cap disks of the MS. Going outwards from the centre, there is the Small Wheel (then the end-cap toroid), the Big Wheel (which contains four layers of devices), and finally the Outer Wheel. There is also an Extra-External (EE) layer in between the Small and Big Wheel.

Structurally, MDTs are akin to the straw tubes that make up the TRT. They are comprised of similar cathode tubes with coaxial anode wires within, but MDTs are filled with mostly Argon gas. As a muon traverses the tube, it ionises the 93% Ar and 7% CO_2 gas, causing free electrons to travel to the inner wire. This creates a small electrical impulse which may be read as a signal that the chamber was hit by a muon. However, unlike the straws in the TRT, the MDTs measure a temporal component to this signal and can infer the radial distance from the inner wire to the ionising muon at its closest approach based on the ion drift velocity. This allows for much higher tracking accuracy with far less tubes. The MDT has a positional accuracy of $25\,\mu\mathrm{m}$ in the bending direction of the magnet. MDTs can reconstruct tracks of muons with $p_T > 4.0\,\mathrm{GeV}$. There are three cylindrical layers of MDTs in the barrel region of the

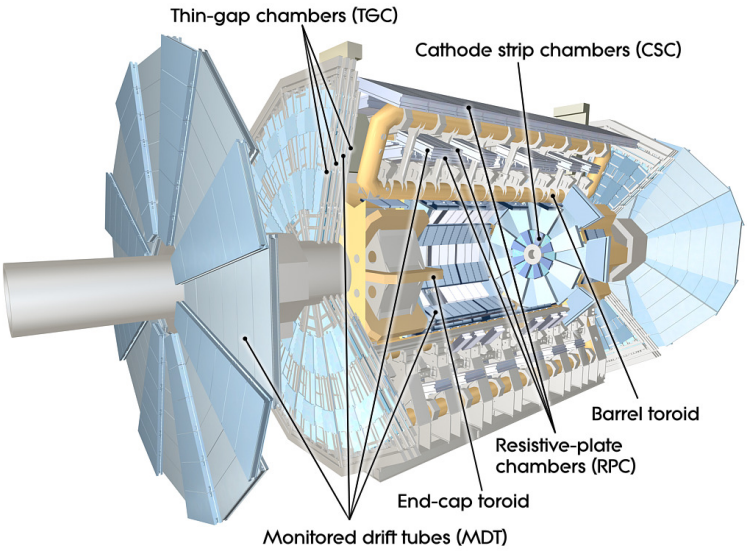


Figure 5.6: The layout of the Muon Spectrometer [192].

MS, providing a coverage of $|\eta| < 1.0$, as shown in Figure 5.7. There are also four MDTs sections in both end-cap regions - a single layer in each of the three wheels, and the EE layer.

CSCs are an example of a multi-wire proportional gas detector. They consist of arrays of positively charged parallel wires, crossing over negatively charge cathode strips. The space between the wires and the strips is filled with a gas made up of 80% Ar and 20% CO_2, and there is a potential difference of 1900 V across it. As a muon ionises the gas, the free positive charges are attracted to the strips, indicating the locations of the hits. The MS uses CSC devices in each of the Small Wheels. Since these are relatively close to the interaction point, they require much faster time resolution and a higher rate capability. The wires are arranged radially and are flanked on either side by cathode layers. On one side the strips are segmented parallel with the wires, and on the other side they are segmented perpendicular to the wires. Therefore, a combination of both layers can provide two coordinates required to calculate the position of the track. The signals from the wires are not read out. The overall resolution is 40 µm in the bending direction and 5 mm in the transverse direction.

For triggering, the MS uses both RPCs and TGCs. The latter of which is very similar to CSCs, with one of the few differences being that the potential of the wires is much higher at 2900 V. There are four layers of TGCs in the end-cap regions, one in the Small Wheel and three in the Big Wheel, covering $1 < |\eta| < 2.4$. RPCs on the other hand, consist of two parallel plates separated by a gas volume 2 mm thick and are only found in the barrel region of the MS. The plates are made from a highly restive material and generate a uniform electric field between them. The pattern of an ion avalanche caused by a muon traversing this gap can give fast measurements of its momentum.

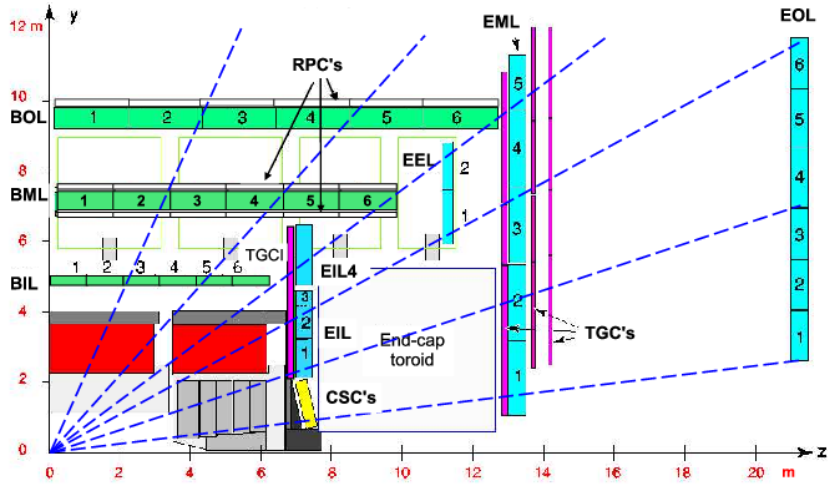


Figure 5.7: Cross-sectional view of the ATLAS Muon Spectrometer in the $r - z$ projection at $\phi = \pi/2$ [193]. The barrel MDT chambers are shown in green and the end-cap MDT chambers are blue. The Small, Big and Outer Wheels are labelled EIL, EML and EOL respectively. The EE layer is also shown.

5.5 The ATLAS Trigger System

This section outlines the trigger system used by ATLAS for data capture during Run 2 of the LHC. A more detailed description is found in Reference [194].

The ATLAS trigger system is as essential as any of the detector's other components and subsystems. Most interesting physics processes have very small cross-sections. Many collisions are required to ensure that they are produced in statistically reliable amounts, yet they still will only make up a small fraction of the number of recorded events. The sole function of the ATLAS trigger system is to make quick and efficient decisions on whether to save data from a particular collision event for further analyses, by checking if the event contained interesting features. This is a crucial step in the experimental process, since saving data from each event to disk will quickly overload the current facilities for long term storage.

The bunch collision rate at the end of Run-2 was 40 MHz. This coupled with the increase in luminosity, pileup, and collision energy over Run-1, was cause for a major upgrade to the trigger system which took place during the LHC Long Shutdown from 2013 to 2014. The new trigger system consisted of a hardware based first level trigger, known as Level-1 [195], and a software based high level trigger [196], referred to as the HLT.

The Level-1 trigger uses only a fraction of the detector's built in electronics to flag and identify Regions-of-Interest (RoIs). As input it takes signals from the dedicated muon triggers, as described in Chapter 5.4, and coarse granularity information from the calorimeters. It searches specifically for signals pertaining to high p_T electrons, muons, photons, and jets. Events which may have high E_T^{miss} also pass the filter. On

average only 1 in 400 events pass the Level-1 trigger, effectively reducing the event rate down to 100 kHz. It takes around 2.5 µs of decision time for a Level-1 accept. Information from further collisions which take place during this time are stored in a short-term memory buffer. If it passes Level-1, the event data and the RoIs are sent to the HLT for further processing.

The HLT is purely software based and has the full granularity information from all detector trackers and calorimeters at its disposal. Yet depending on why the event passed the Level-1 trigger, the HLT might only investigate within the RoIs. This reduces decision time taken to look at superfluous detector signals. The HLT employs sophisticated reconstruction algorithms which were optimised during the shutdown to better reflect those which may be used in offline analyses. The average output rate of the HLT to long term storage is around 1 kHz, though this is primarily due to the speed at which events can be processed offline, as well as limitations in the total storage capacity.

5.6 Pileup

In each bunch-crossing at ATLAS there is usually only a single pp interaction which is of interest. This is called the hard-scatter. However, reconstruction of this specific interaction is obscured by overlapping signals from additional pp interactions. This phenomenon is referred to as pileup.

There are two types of pileup [197]. The first is in-time pileup, and it is unavoidable at the LHC. As explained in Chapter 4, the proton beams at the LHC consist of many discrete bunches of protons. Each contain around 115 billion protons and, at the interaction point in the centre of ATLAS, two bunches collide every 25 ns. This feat leads to the LHC's impressively high luminosities, but it is a double-edged sword, for each bunch-crossing may give rise to multiple pp interactions. Particles which are produced in these additional pileup interactions overlap with those produced during the hard-scatter. The number of inelastic pp collisions which take place during a bunch-crossing μ can be described by a Poisson distribution.

$$\mu = \frac{\mathscr{L}\sigma_{pp}}{N_p f}, \tag{5.4}$$

where $\mathscr{L}$ is the instantaneous luminosity, σ_{pp} is the cross-section for inelastic pp interactions (which at $\sqrt{s} = 13$ TeV is around 80 mb), N_p is the number of protons per bunch and finally f is the revolution frequency of the bunch around the LHC*. The number of interactions for the 2017 Run-2 data, the data used in this dissertation, is shown in Figure 5.8.

*The reason that μ seems to be inversely proportional to both N_p and f is that these terms already exist in the definition of $\mathscr{L}$ shown in Equation 4.2.

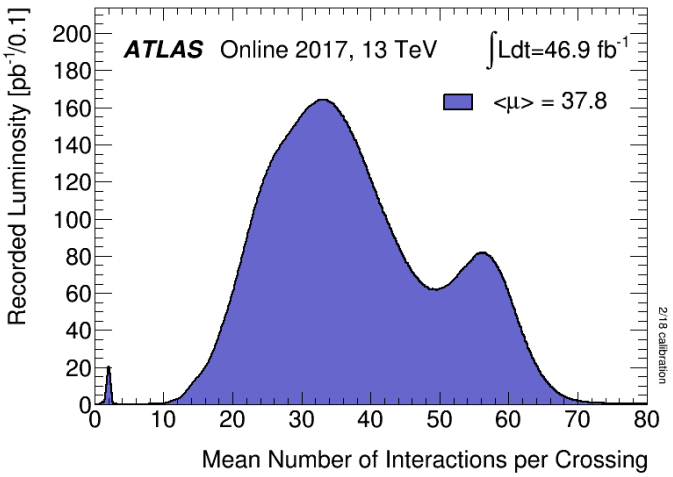


Figure 5.8: The luminosity-weighted distribution of the mean number of interactions per crossing for the 2017 pp collision data at $\sqrt{s} = 13$ TeV [198].

During event reconstruction, Chapter 6, significant effort is expended to identify and categorise which objects came from the hard-scatter versus a pileup pp collision. This task is challenging but feasible for charged particles which leave tracks in the ID and can therefore be linked to a scatter location via vertex tagging, but it is extremely onerous for neutral objects which leave no tracks at all. In-time pileup significantly degrades measurements of object energies within the calorimeters. It has a large effect on lepton isolation energies, jets, and the identification accuracy of electrons.

Arguably the most significant effect, especially in the context of this dissertation, is the degradation that in-time pileup causes to the measurement of missing transverse momentum. This has widespread consequences in many physics analyses. E_T^{miss} reconstruction is particularly sensitive to pileup as it utilises measurements taken from all detector subsystems and requires the most unambiguous representation of the hard-scatter interaction. Much of the work done in the past few years by the ATLAS Jet/E_T^{miss} performance group has focused on developing methodology that is resilient to an increase of pileup interactions (see Chapter 7 for more details). Maintained quality of E_T^{miss} reconstruction under high in-time pileup conditions was one of the main goals of this work.

Out-of-time is the other type of pileup, and it is slightly more manageable. It is an occurrence when signals from previous bunch-crossings interference with the current one. This is primarily an issue in the ECal which has a long signal shaping time. The sensitivity window of the ECal is longer than the 25 ns bunch spacing, and some of the subsystems are not able to fully reset before the next set of collisions.

Chapter 6

Event Reconstruction

This chapter summarises the event reconstruction procedure used by the ATLAS Collaboration. This process involves collecting and converting the basic signals and outputs of the many sub-detectors of ATLAS into collections of calibrated objects with well-defined properties. Event reconstruction is performed by standardised algorithms shared across the ATLAS experiment. There are several layers to this process which are completed in sequence, whereby the output of one layer is passed on to the next. The layers combine to achieve more sophisticated reconstructions and deeper levels of understanding.

Two of the more basic steps in event reconstruction are track finding and the formation of calorimeter clusters. The tracks and clusters are then used for the identification and calibration of particles such as electrons, muons and photons. Jets, which are collections of fast-moving particles produced by QCD fragmentation, are also reconstructed from clusters. Intermediate particles such as tau leptons and b-quarks are identified using the outputs of the track, particle and jet reconstruction processes. Higher-level variables such as E_T^{miss} are created from the set of calibrated particles, jets and other signals.

In most papers, E_T^{miss} reconstruction is included together with all other features of the ATLAS event reconstruction. However, since E_T^{miss} is the focus of this dissertation, the associated algorithms and methods are given a more in-depth overview in Chapter 7.

Tracks and vertices

One of the first layers of event reconstruction is track finding [199, 200]. This is done by requiring a minimum number of hits in the ID and the MS [†]. The track is then extended using either a Gaussian sum filter or a χ^2 fit procedure to include more discrete hits. Tracks in the ID, which should be helical due to the passage of a charged particle moving through the axial magnetic field, are then fitted with several variables: the track's radius of curvature, polar and azimuthal angles, as well

[†]The minimum number of required hits to initiate a track can change depending on the desired quality

as the transverse and longitudinal impact parameters which are labelled d_0 and z_0 respectively. The measurement of d_0 is the minimum distance between the track and the beam line, while z_0 is defined as the distance between the track and the interaction point in the longitudinal direction at its closest approach. A track is accepted for later use in reconstruction if $p_T > 400\,\text{MeV}$ and $|\eta| < 2.5$.

Vertices are created from the convergence of multiple tracks near the beam line, indicating the location of an underlying physics process and the subsequent emission of charged particles [201]. Due to in-time pileup and the decay of particles with short lifetimes, multiple vertices may be reconstructed for a single bunch-crossing. Several of these reconstructed vertices are identified as primary vertices which potentially show the location of a pp scatter.

The primary vertex of the hard-scatter interaction, labelled PV_0, is defined as the one with the largest $\sum(p_T^{\text{track}})^2$. All other primary vertices are assumed to have been produced by in-time pileup. The total number of reconstructed primary vertices in a single bunch-crossing is N_{PV}. This variable is strongly correlated to μ, the number of expected inelastic pp interactions given the luminosity of the colliding bunches.

Cluster

In addition to tracks, the other type of basic input to particle identification algorithms at ATLAS is the calorimeter cluster. Cluster reconstruction involves grouping energy readings from adjacent calorimeter cells so that a cluster may encapsulate the full shape of a particle shower. The combined energy of a cluster can be used to estimate the energy of the incident particle. Electrons and photons tend to produce much more narrow particles showers in the ECal, compared to particles that interact hadronically, such as pions and kaons. These usually lead to wider and deeper showers which penetrate through to the HCal. Cluster reconstruction is performed in both the ECal and the HCal, and can span the boundary between them. Two types of clustering algorithms are used in ATLAS [202]. A sliding-window algorithm, which sums cells within a fixed-size rectangular window, is primarily used to reconstruct electrons and photons. For the reconstruction of jets, a topological algorithm [203] is used whereby clusters begin with seed cells, which are then grown iteratively outwards if neighbouring cells contain energy above predefined noise thresholds.

6.1 Particle Reconstruction

This section describes the higher-level reconstruction of particles and physics objects in ATLAS using tracks and clusters. A cut-away image of the various ATLAS subdetectors is shown in Figure 6.1, and it illustrates how they record the passage of various types of particle.

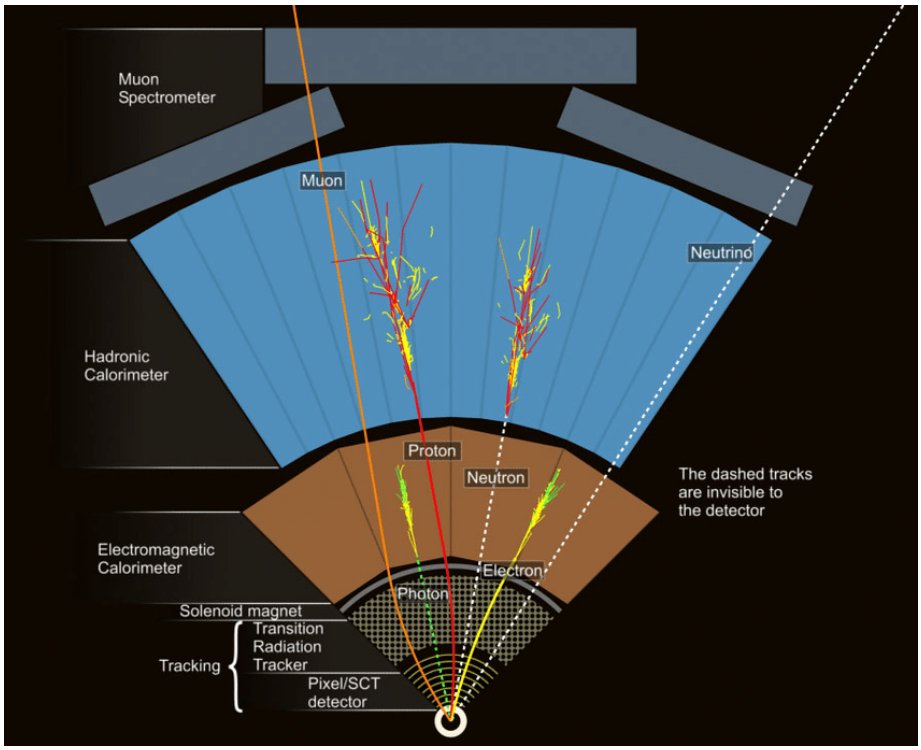


Figure 6.1: A schematic cut-away of the ATLAS detector. The characteristic signatures of various particles traversing the detector are shown [204].

For each physics object in ATLAS, there exist varying levels of particle identification (PID) criteria. Each PID operating point has well-defined signal efficiencies and background rejection rates that have been measured in data and/or simulation. Usually, three operating points exist for each object: Loose, Medium and Tight. The Medium identification criteria is usually the default, providing a balance between efficiency and signal purity. The Loose criteria is set to maximise efficiency at the price of quality, and the Tight criteria is set to maximise background rejection at the price of efficiency. This nomenclature is used throughout ATLAS when describing most variable cuts, requirements, selections or filters. Though, as encountered further on in this chapter, there are exceptions to this naming scheme. Each subsequent operating point is nominally designed to produce a subset of objects passing the operating point preceding it. Therefore, objects meeting Tight PID criteria must also satisfy Medium requirements, and those selected by a Medium cut are also selected by a Loose cut.

In addition to PID, leptons and photons often must satisfy isolation requirements. An isolated object is one that is surrounded in (η, ϕ)-space by little other detector activity. This typically reduces the inclusion of objects which were not emitted by the hard-scatter and were perhaps the result a hadron decay or a misidentified energy deposit. The efficiency of an isolation requirement ϵ_{iso} is defined as the ratio between the number of true hard-scatter objects passing the requirement to the total number of true hard-scatter objects passing the default PID criteria. Three main

methods are used to create isolation requirements at ATLAS [205]. The first methods is called FixedCut, which places hard energy limits on the amount of detector activity surrounding the object. The second method, called Gradient isolation, allows the limits of surrounding activity to change in order to target a value of ϵ_{iso} that is uniform in η. These two methods have additional operating points, such as Gradient-Loose (less strict definition of Gradient) and FixedCut-Tight (more strict definition of FixedCut). The final method targets a specific value of ϵ_{iso} that is uniform in both η and the object's transverse energy E_T. This final method is the default and it thus only labelled by its operating point, such as Tight or Loose isolation.

6.1.1 Muons

Muons are the only detectable object consistently capable of traversing the entire detector. Muons leave tracks in both the ID and the MS as shown by Figure 6.1. They usually leave little energy the calorimeters due to their minimum-ionising behaviour. By design, the ATLAS detector terminates all other interacting particles in its calorimeters, so muons can be identified simply by the fact that a signal was detected in the MS. The muon p_T is calculated from track curvature due to the ATLAS solenoid and toroidal magnet systems with a correction for small energy losses in the calorimeters. At ATLAS, four different and complimentary types of muon reconstruction algorithms are utilised, each using different information from the sub-detectors [206]. The different types of reconstructed muon are shown in Figure 6.2.

- **Combined muons**: Track reconstruction is carried out independently in the ID and the MS. For each MS track, a partner search begins for a corresponding track in the ID. The partners are checked for momentum compatibility and matching χ^2, accounting for the magnetic fields in the detector. MS hits may be added or removed from the track in order to improve the global fit quality. The combination of both sub-detector readings leads to optimal momentum resolution and background rejection [207]. This is the most reliable method for reconstructing and tagging a muon.

- **Stand-alone muons**: The muon trajectory is reconstructed purely from an MS track. This is often the case when a muon is emitted outside the coverage of the ID, $2.5 < |\eta| < 2.7$. The track parameters are loosely checked for compatibility with originating from the IP, but without information from the ID these muons suffer reduced momentum and impact parameter resolution. They are also referred to as extrapolated track muons.

- **Segment-tagged muons** A muon with low transverse momenta can undergo multiple scatterings in the detector, leading to an unmatched MS track in the innermost layer of the MS only. A segment-tagged muon's kinematics are taken purely from an ID track, if it can be associated with an incomplete MS

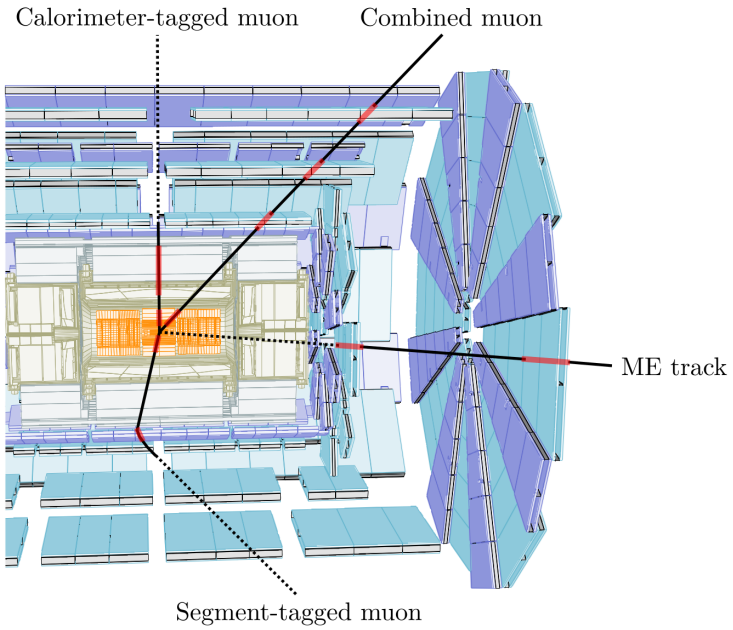


Figure 6.2: A visualisation of the different reconstructed muon types found at ATLAS [209].

track segment. Further requirements are placed on the segments to reduce the contribution of secondary muons produced from meson decays [208].

- **Calorimeter-tagged muons**: A muon can be reconstructed if an ID track is matched to energy clusters in the calorimeters consistent with a minimum-ionising particle. Due to the lack of hits in the MS, this type of muon suffers from high background contamination. This type of muon exists mostly for regions where there is poor coverage by the MS, such as around $|\eta| \approx 0$.

There are four different PID operating points applied to reconstructed muons, labelled Loose, Medium, Tight, and High-p_T [206]. These criteria are based on the type of muon, its kinematics and other variables. The Medium identification criteria accepts only combined or extrapolated track muons and minimises systematic uncertainties associated with reconstruction. The High-p_T identification criteria is applied only to combined muons passing Medium PID. This provides the best resolution for muons with transverse momenta above 100 GeV.

An isolation requirement is typically placed on reconstructed muons, which suppresses non-prompt muons typically produced by meson and heavy-flavour semi-leptonic decays. Most commonly used for muons is the Gradient isolation operating point, which is defined so that ϵ_{iso} is at least 90% for muons with $p_T > 25$ GeV and 99% at 60 GeV. Muon reconstruction and isolation efficiencies were measured in data and simulation using a large sample of $J/\Psi \to \mu\mu$ and $Z \to \mu\mu$ decays [206].

6.1.2 Electrons

The other type of cleanly reconstructed lepton at ATLAS is the electron. Electrons leave a curved track in the ID and terminate with a particle shower in the ECal. Therefore, they are only reconstructed in the central region of the ATLAS detector with a veto in the transition region between the barrel and the end-cap calorimeters ($1.37 < \eta < 1.52$).

The first step in electron reconstruction [205] is the creation of clusters in the ECal. All three layers of the ECal are combined to form discrete energy towers of $\Delta\eta \times \Delta\phi = 0.025 \times 0.025$. A sliding-window algorithm [202] of size 3×5 towers is then used. The position of the window is adjusted until the contained transverse energy is a local maximum. To optimise reconstruction efficiency while minimising contribution from electronic or pileup noise, the transverse energy must also exceed 2.5 GeV. If this threshold is met, then the region is marked as a seed-cluster. Duplicate removal is then carried out for overlapping seed-clusters [202].

For each seed-cluster, an algorithm searches for a matching track in the ID. If multiple matching tracks are found, then the one with the smallest ΔR to the centre of the cluster is selected. The impact parameters of the matched track are then checked to see if they are consistent with a primary vertex. Fully reconstructed clusters are formed around the seed-clusters by enlarging their sizes to 3×7 units in the barrel and 5×5 units in the endcap region of the calorimeter [205]. This is to ensure the capture of the full electron energy, including energy lost due to bremsstrahlung radiation. The final four-momentum of the cluster is then calibrated using the energy deposited, and the track kinematics.

This type of signature suffers from large backgrounds. For example: a charged pion can be mistaken as an electron since it leaves a similar track in the ID. Identification of signal electrons versus background is established on a set of track-based and calorimeter-based variables. These include results from the TRT, as electrons generate much higher transition radiation than heavier particles like pions. The full list of inputs used in this multivariate analysis (MVA) is found in Reference [210]. Electron identification is then based on a requirement of a single value; the ratio of the signal to background likelihood function. This is known as likelihood-based (LLH) identification.

As was the case with muons, there are different levels to the electron identification criteria and in order of increasing background rejection they are labelled as LooseLLH, MediumLLH, and TightLLH. These operating points use the likelihood discriminant but with a different cut value. To further suppress the contribution from non-signal electrons, isolation requirements are applied. Electron isolation is determined by two variables. The first is the total transverse momentum of all tracks emerging from PV_0 which lie within $\Delta R = \min(0.2, \frac{10\,\text{GeV}}{p_T^e})$ of the electron direction,

where p_T^e is the transverse momentum of the electron. The second variable is the total transverse energy measured in all calorimeter cells in a cone of $\Delta R = 0.2$ around the candidate electron. Gradient isolation is defined on electrons with the same targeted efficiencies as with muons, measured using the tag-and-probe method on $J/\Psi \to ee$ and $Z \to ee$ events [210].

6.1.3 Photons

Other than the lack of a track in the ID, the experimental signature of a photon produced in a pp collision at ATLAS is very similar to that of an electron. They therefore can only be identified by their electromagnetic shower and associated cluster in the ECal. The signature of such a photon is shown in Figure 6.1.

It is possible that a high energy photon underwent e^+e^- pair production in the detector material before reaching the ECal. These photons are described as being converted. The oppositely charged electrons produced by photon conversion will create tracks in the ID which emerge from a vertex displaced from the interaction point. Therefore, the reconstruction procedure of a photon matches that of an electron until the point where the seed cluster is checked for a matching track in the ID. If the matching fails, then the cluster is tagged as an unconverted photon. If a matching is possible but the track does not emerge from a primary vertex, then the cluster is tagged as a converted photon. It is important to note that around 30% to 35% of reconstructed photons at ATLAS are converted [211].

Photons are also produced from charged particles emitting bremsstrahlung radiation as they move through the detector. To identify the unconverted photons more likely to be produced by the hard-scatter, PID and isolation criteria must be met. For data collected at $\sqrt{s} = 13\,\text{TeV}$ only two reference sets of PID cuts are defined, Loose and Tight [212]. Identification is based on shower shapes in the second layer of the ECal and the amount of energy deposited in the HCal. The most commonly adopted isolation requirements for final-state photons are also labelled Loose and Tight. They each involve predefined limits on the total transverse momenta from all ID tracks and the sum of all transverse energy contained in calorimeter clusters within a cone surrounding the photon. For the Loose operating point, the cone has a width of $\Delta R = 0.2$, which is increased to $\Delta R = 0.4$ for Tight. The track and calorimeter limits for both operating points are directly proportional to the measured photon transverse momentum.

Photon identification and isolation efficiencies are measured in data using radiative Z decays and electron extrapolation and differ when dealing with converted versus unconverted photons [212].

6.1.4 Jets

The reconstruction of jets is crucial at ATLAS as they are often the only way to infer the production of a gluon or a quark in a physical interaction. Due to QCD confinement, high energy quarks and gluons emitted from a pp collision pull additional coloured objects from the QCD vacuum. These coloured objects hadronise into colourless bound states before they reach the detector. The result of this process is a collimated spray of hadrons known as a jet. Since some of these hadrons may be charged, many tracks will be left in the ID and large energy deposits will also be found in both the ECal and the HCal. The measured energy and direction of a jet provides information about the hadronic (coloured) energy flow produced in a collision.

At ATLAS jets are reconstructed using three-dimensional topological clusters of energy deposits in the calorimeters called topoclusters [202]. Clustering methods attempt to regroup the many particles and signals in the detector into a single four-vector representing the initial energy and momentum of the hard-scatter parton. A jet finding algorithm [202] begins by identifying the calorimeter cells that contain the most significant energy deposits which must be at least four times higher than the expected electronic noise. These cells are used as starting points to construct the topocluster which is grown iteratively outward by including adjacent calorimeter cells provided they contain signals greater than twice the expected noise. Once all signals have been collected into different topoclusters they are each attributed with a position, taken by an energy weighted average over each cell in the topocluster, and an energy measurement, taken by the total energy in all cells that make up the cluster [213].

At ATLAS, the anti-k_t algorithm [203] is used to combine topoclusters into a single candidate jet. It begins with the highest p_T topocluster in the event and combines it with all others within a predefined radius R which satisfy certain criteria. If the clusters are combined, then so too are their four-momenta and another search within a radius R of this new vector is performed. At the end of this iterative procedure all topoclusters should have been combined into or replaced with candidate jets each with a single attributed jet four-vector.

Jet Calibration

Once jet candidates are created by the anti-k_t algorithm, they are calibrated to better reflect the energy and momenta of the initial partons at a particle level. Jet calibration involves a sequential scheme of corrections which are derived from both MC simulation and data. The first step is known as origin-correction and this process recalculates the jets' four-vectors. Initially they had been constructed to point outward from the geometrical centre of the detector and the correction moves their origin to PV_0. This improves the angular resolution of the reconstruction. ID tracks are also

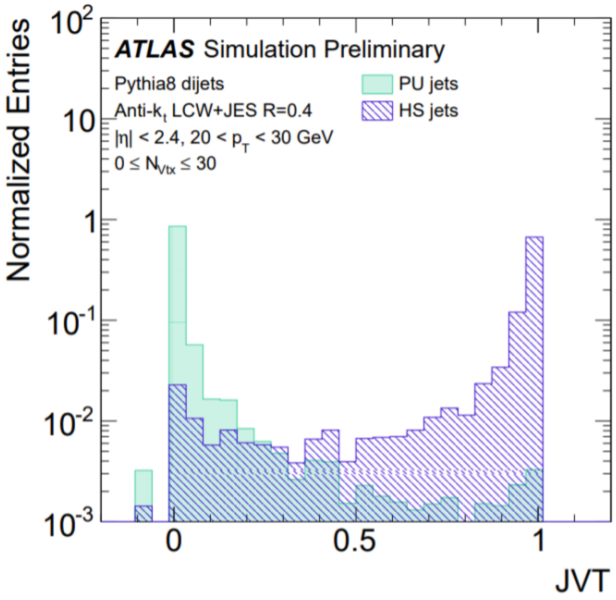


Figure 6.3: The distribution of the JVT score for hard-scatter jets and pileup jets with $20 < pT < 30\,\text{GeV}$ in simulated dijet events [217]. Jets which have no associated tracks are given a JVT score of -0.1.

matched to each jet via a process called ghost-association [153]. These tracks must have $p_T > 500\,\text{MeV}$ and had to have emerged from a primary vertex.

The p_T of each jet is corrected to account for the additional energy deposited within the jet radius from both in-time and out-of-time pileup [214]. This correction considers N_{PV} and μ. Next, the jet energy scale (JES) correction is applied to calibrate the reconstructed jet energies at the electromagnetic scale to the true energy scale at particle level. This considers different detector responses, reconstruction inefficiencies and the non-compensation of hadronic calorimeters. Determination of the JES and the jet energy resolution is based on MC simulations of QCD dijet events [214]. Systematic uncertainties associated with the JES are often the most significant sources of experimental uncertainties at the ATLAS experiment. Final data driven in-situ calibrations are applied to correct for MC mismodeling in the previous calibration step [215, 216].

Jet Vertex Tagger

In-time pileup activity often creates jets which are not involved in the hard-scatter and are thus background. To discriminate between signal and pileup jets, ATLAS currently uses an algorithm called the Jet Vertex Tagger (JVT) [217]. Figure 6.3 shows the distribution of the JVT discriminant output for jets originating from the hard-scatter and for those from pileup. The JVT operating points were evaluated on the full 2015 and 2016 datasets by selecting a sample of Z boson events decaying into muons with at least one extra jet. The recommended Medium JVT operating point corresponds to an average efficiency of 92% [217] for jets with $p_T < 60\,\text{GeV}$.

A limitation of the JVT discriminant is that it can only be used for jets which are adequately covered by the ID. Therefore, jets are only required to pass a JVT check if $|\eta| < 2.5$. However, jets at ATLAS can be reconstructed up to $\eta = 4.5$ due to the forward calorimeters. The rejection of pileup jets in this forward region is crucial to enhance the sensitivity of many analyses. For this reason, the forward-Jet Vertex Tagger (fJVT) [218] was developed. This is another multivariate discriminant which takes as input the projections of the associated track p_T along the forward-jet's transverse momentum, amongst other variables. In this analysis any jet with $\eta < 2.5$ is classified as a forward-jet, while the others are referred to as central-jets.

b-tagging

Jets which arise from the hadronisation of a b-quark have distinguishing features from those which originated from lighter partons. Since they decay primarily due to the weak force, hadrons formed from b-quarks are relatively long lived and in ATLAS the decay lengths are of the order of millimetres, a distance large enough to be resolved by the ID. Therefore, jets from b-quarks usually have tracks originating from displaced secondary-vertices. Identifying such jets in an event is a process called b-tagging. Several techniques have been developed for b-tagging at ATLAS which utilise different information to search for evidence of b hadron decay. The IP3D algorithm [204] uses impact parameters of the tracks associated with jets. The SV1 algorithm [204] is a likelihood discriminant using variables such as the invariant mass of all tracks. The JetFitter algorithm [219] makes use of the topological structure of b- and c-hadron decays to reconstruct the decay chain inside the jet. The current method for b-tagging is the MV2c10 algorithm [220, 221]. This is a BDT which combines all methods into a single score and was trained on simulated $t\bar{t}$ events.

6.2 Object Selection

This section describes the specific reconstruction criteria applied to each type of physics object used in this dissertation. Generally, these requirements differ between physics analyses. The set of criteria presented here is therefore only one example and are based on those used in the ATLAS papers in References [9, 222]. Furthermore, they are consistent with the default ATLAS selections used across all SUSY searches for electroweak superpartners.

Since current methods of E_T^{miss} reconstruction are object based, as will be explained in Chapter 7, E_T^{miss} reconstruction performance must always be provided in the context of the chosen selection criteria used to define those physics objects. This is especially relevant for the results presented in this dissertation, since many of the input variables used to train the neural networks are dependent on the object selection as will be detailed in Chapter 9. Therefore, the specifics of the object selection criteria restricts the applicability of the network. Using any other selection criteria without

retraining the network would fundamentally change the way it interprets an event. Due to the dependence of the network on these criteria, each object selection process is presented here in full detail. The same criteria were applied to all samples throughout this dissertation.

Two selections of objects are defined which are referred to as baseline and signal. These selections impose cuts based on variables such as the objects' kinematics, PID quality and (if it is a lepton or a photon) isolation. Objects passing baseline selection are used as inputs for the overlap removal (OR) procedure, described in Section 6.3, as well as inputs for several of the E_T^{miss} reconstruction algorithms, detailed in Chapter 7. Signal objects, and only those which pass OR, are used to construct multiplicity and kinematic discriminating variables needed for event selection. Both baseline and signal objects are used to define some of the inputs for the neural network as explained in Section 9.4. Signal requirements are always equivalent to or stricter than baseline so the collection of signal objects in a given event will be a subset.

The object selections used in this analysis for electrons, muons, photons, and jets are found in Tables 6.1, 6.3, 6.2 and 6.4, respectively. It is important to note that the reconstruction of hadronic taus was not performed in this analysis, in line with the procedures used in the aforementioned papers. Hadronically decaying taus are simply left as calibrated jets.

Electrons

All reconstructed electrons were required to have been detected within $|\eta| < 2.47$. There was an additional veto on electrons which traverse the transition region between the barrel and end-cap electromagnetic calorimeters ($1.37 < |\eta| < 1.52$) as energy deposited by electrons in this region would likely be reconstructed as a jet if it met the corresponding selection criteria. Baseline electrons had to have $p_T > 10\,\text{GeV}$ and satisfy the LooseLLH PID quality benchmark. The longitudinal impact parameters of baseline electrons were also required satisfy $|z_0 \sin\theta| < 0.50\,\text{mm}$*. To be categorised as a signal electron, baseline electrons needed $p_T > 20\,\text{GeV}$ and to have satisfied both the Gradient-Loose isolation and MediumLLH PID criteria. For electrons with $p_T > 400\,\text{GeV}$ the isolation requirement was changed to a FixedCut method, which placed a 3.5 GeV limit on all energy deposited in the calorimeters within $\delta R = 0.2$. Signal electrons also had criteria placed on the significance of their transverse impact parameter, $|d_0/\sigma_{d_0}| < 5$.

*Since pileup vertices are nominally distributed along the z-axis, the longitudinal impact parameter is the optimal discriminating feature for objects which leave tracks. It is thus applied at baseline level definitions while transverse impact parameter is only used later in the selection process

Feature	Requirement
Baseline Electron	
Geometric Acceptance	$\lvert\eta\rvert < 2.47$ and not $1.37 < \lvert\eta\rvert < 1.52$
Kinematic Acceptance	$p_T > 10\,\text{GeV}$
PID Quality	LooseLLH
Impact Parameter	$\lvert z_0 \sin\theta\rvert < 0.50\,\text{mm}$
Signal Electron	
Kinematic Acceptance	$p_T > 20\,\text{GeV}$
PID Quality	MediumLLH
Isolation	Gradient-Loose
Impact Parameter	$\lvert d_0/\sigma_{d_0}\rvert < 5$

Table 6.1: Summary of the electron selection criteria. The signal selection requirements were applied on top of the baseline selection.

Photons

Baseline photons were required to have $p_T > 25\,\text{GeV}$ as well as $|\eta| < 2.37$. As was the case with elections, the transition region of $1.37 < |\eta| < 1.52$ was excluded. Both signal and baseline photons had to meet Tight identification criteria. The only additional filter applied by the signal selection was the isolation requirement matching the Tight operating point, which placed separate limits on the total transverse momentum of all tracks and all calorimeter deposits within a cone of $\Delta R = 0.4$ around the photon.

Feature	Requirement
Baseline Photon	
Geometric Acceptance	$\lvert\eta\rvert < 2.37$ and not $1.37 < \lvert\eta\rvert < 1.52$
Kinematic Acceptance	$p_T > 25\,\text{GeV}$
PID Quality	Tight
Signal Photon	
Isolation	Tight

Table 6.2: Summary of the photon selection criteria. The signal selection requirements were applied on top of the baseline selection.

Muons

As with electrons, all muons used in this analysis were required to have $p_T > 10\,\text{GeV}$ and were restricted to the same η range of $|\eta| < 2.47$, though without the veto in the calorimeter transition region. The Medium PID quality criteria was applied to both baseline and signal muons, though signal muons, as with the case with electrons, must have also satisfied Gradient-Loose isolation. A cut on the longitudinal impact parameter, $|z_0 \sin\theta| < 0.50\,\text{mm}$ was applied at a baseline level, while a cut on the transverse impact parameter significance was only applied to signal muons, $|d_0/\sigma_{d_0}| < 3$.

Feature	Requirement
Baseline Muon	
Geometric Acceptance	$\lvert\eta\rvert < 2.47$
Kinematic Acceptance	$p_T > 10\,\text{GeV}$
Impact Parameter	$\lvert z_0 \sin\theta\rvert < 0.50\,\text{mm}$
Signal Muon	
Kinematic Acceptance	$p_T > 20\,\text{GeV}$
PID Quality	Medium
Isolation	Loose Gradient
Impact Parameter	$\lvert d_0/\sigma_{d_0}\rvert < 3$

Table 6.3: Summary of the muon selection criteria. The signal selection requirements were applied on top of the baseline selection.

Jets

In this analysis there are several different selections of jets on top of the standard baseline and signal. The selection of jets used to calculate E_T^{miss} has a large impact on its performance. Several of the object-based E_T^{miss} algorithms described in Chapter 7 differ only because they were constructed using different collections of jets. The jet collections are labelled Loose, Tight and FJVT. These three collections of jets are not orthogonal. Both the FJVT and Tight collection of jets are a subset of the Loose collection, which is in turn a subset of all baseline jets. Each alternative jet collection gives rise to a different E_T^{miss} reconstruction which shares its name. Signal jets which are strictly used for event selection and not E_T^{miss} reconstruction, follow the same selection criteria as Reference [222] and are a subset of FJVT jets. In addition, b-tagging is performed on all central baseline jets creating another subset. To present the many selections for the various collections of jets in a coherent manner, the cuts are documented below in the order that they were applied during the processing of a given event.

Jets were first reconstructed using topoclusters and the anti-k_t algorithm [203] with a distance parameter of $R = 0.4$. They were then checked if they met baseline requirements which were defined to be only $p_T > 20\,\text{GeV}$ and $|\eta| < 4.5$.

The collection of baseline jets were then filtered to create the Loose set by removing jets likely originating from pileup interactions. This was achieved by applying a cut based on the Medium JVT operating point. Jets automatically pass this cut if $p_T > 60\,\text{GeV}$ or $|\eta| > 2.5$. All jets with $|\eta| < 2.4$ were required to have $JVT > 0.59$, while jets within $2.4 < |\eta| < 2.5$ needed only to have $JVT > 0.11$ (the lower value was set to cope with the lower efficiency in this region). Baseline jets passing this JVT cut were immediately saved as Loose jets.

The Tight collection of jets is a subset of Loose but was created by removing all forward-jets with $|\eta| > 2.5$ and $p_T < 30\,\text{GeV}$, as this region of phase space had been measured to contain more pileup jets than hard-scatter jets [9]. The FJVT jet

collection was also derived from Loose jets, but used an alternative approach to suppress forward pileup jet contamination. The original threshold for forward-jet p_T of 20 GeV was kept, but all Loose forward-jets with $p_T < 50$ GeV that failed the fJVT-Loose criteria were removed, as defined in Reference [218].

Signal jets are a subset of FJVT jets, with an additional cut on $|\eta| < 2.8$. All baseline jets with $|\eta| < 2.5$ were also passed through a b-tagging procedure based on the MV2c10 algorithm. The chosen b-tagging operating point corresponded to a 77% average efficiency measured using simulated $t\bar{t}$ events [221]. The collection of all b-tagged jets satisfying signal requirements was also saved.

Feature	**Requirement**
Baseline Jet	
Collection	Anti-k_t, $R = 0.4$
Geometric Acceptance	$\|\eta\| < 4.5$
Kinematic Acceptance	$p_T > 20$ GeV
Loose Jet	
Collection	Baseline jets
JVT	Medium
Tight Jet	
Collection	Loose jets
Forward-Jet Restriction	$p_T > 30$ GeV
FJVT Jet	
Collection	Loose jets
Forward-Jet Restriction	fVJT-Loose or $p_T > 50$ GeV
Signal Jet	
Collection	FJVT jets
Geometric Acceptance	$\|\eta\| < 2.8$
Signal b-Jet	
Collection	Signal jets
Geometric Acceptance	$\|\eta\| < 2.5$
b-tagger Algorithm	MV2c10
Efficiency	77%

Table 6.4: Summary of the jet selection criteria. The multiple and overlapping collections of jets exist to give rise to different estimations of E_T^{miss}. The first row labelled "collection" lists the previous set of jets used to derive the current one; for baseline jets this corresponds to the output of the Anti-k_t algorithm. The requirements of Tight and FJVT jets are applied to forward-jets only, which are defined as those which have $|\eta| > 2.5$.

6.3 Overlap Removal

It is possible for two or more objects to overlap in (η, ϕ)-space. Where this occurs, only one of the objects is considered while the other is rejected. This process is referred to as overlap removal (OR) and it is a necessary step to prevent the double counting of detector signals. It is important to note that in this dissertation there are two very similar procedures which attempt to prevent double counting. Both are performed primarily on baseline objects but differ in function.

The first procedure is referred to simply as OR and it is used to decorate physics objects. Only baseline muons, electrons, photons, and jets, that pass both OR and signal selections are used to create the main discriminating variables for analyses. Several operating points exist for OR, but they all prioritise the collection of clean and calibrated objects from independent signals. A second but similar process is referred to as signal ambiguity resolution and it concerns E_T^{miss} reconstruction. It exists to prevent the double counting of momenta. This process is much stricter than OR and prioritises the collection of all independent sources of transverse momenta rather than high quality prompt objects. It is carried out independently to OR and it only affects the measured E_T^{miss} of an event. Signal ambiguity resolution is covered in Chapter 7.

The particular OR configuration used in this analysis corresponded to the boosted-lepton operating point [223]. OR is performed on the set of all baseline muons, electrons, photons and jets and involves the following steps performed in listed order. Only surviving objects participate in subsequent steps.

- Any electron which shares a track with a higher p_T electron is removed. This can occur if two different seed clusters are associated with the same track.
- Any electron is removed if it shares an ID track with a muon. However, if it is a calorimeter-tagged muon then the electron survives, and the muon is rejected.
- Any photon is removed if it is measured to be within $\Delta R < 0.4$ of a lepton.
- Both electron and jet candidates are created from clusters in the ECal, therefore electrons will always be reconstructed as a jet. If a jet is within $\Delta R = 0.2$ of an electron, then it is discarded since it likely originates from the electron induced shower.
- Electrons within a sliding window defined by $\min(0.4, 0.04 + \frac{10\,\text{GeV}}{p_T^e})$ of a remaining jet are removed to suppress electrons emitted from semileptonic decays of b- and c-quarks.
- Jets which have less than three tracks are discarded if they overlap with a muon candidate that carries a significant fraction of the jet transverse momentum ($p_T^\mu > 0.7 \sum p_T^{\text{jet tracks}}$). In this step the muon is deemed to overlap with the

jet if either it is within $\Delta R < 0.2$ or if the muon is matched to a track associated with the jet. This is to remove jets which may have originated due to bremsstrahlung radiation emitted from the muon as it traverses the calorimeters.

- Muons within a sliding window defined by $\min(0.4, 0.04 + \frac{10\,\text{GeV}}{p_T^\mu})$ of a remaining jet are removed to suppress muons emitted from semileptonic decays of b- and c-quarks.
- Finally, any photon is rejected if they are measured to be within $\Delta R < 0.4$ of a jet which also passes the JVT requirements.

Chapter 7

Current $E_{\mathrm{T}}^{\mathrm{miss}}$ Reconstruction at ATLAS

This chapter describes the current state of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction at ATLAS. Information presented here is a summary of the ATLAS public papers found in References [9–11].

Over the past few years, ATLAS has employed several algorithms to reconstruct the missing transverse momentum of the hard-scatter. These different algorithms, referred to as $E_{\mathrm{T}}^{\mathrm{miss}}$ working points in this document, are alike in that they attempt to estimate the transverse momentum carried away by undetected particles, a two-dimensional vector in the $x - y$ plane of the ATLAS detector. This is done by using the negative vectorial sum of the transverse momenta of observed particles in the final state. The working points differ primarily in one of two ways. Some differ in the information used to reconstruct the p_{T} of the final state particles: using either tracks in the ID, energy deposits in the calorimeters, fully calibrated physics objects, or a combination of all three. In the case where fully calibrated objects are used, the second discerning feature between the working points is the selection criteria for which objects participate in the vector sum. Ideally the vector sum should include all detected particles that were deemed to have emerged from the hard-scatter while excluding all those that were emitted by pileup interactions. But perfect separation of these signals is impossible, and the different working points offer varying levels of signal efficiency and pileup suppression.

Throughout this document, five different $E_{\mathrm{T}}^{\mathrm{miss}}$ working points are compared, contrasted, and used as inputs for the neural networks in Chapter 9. This is not an exhaustive list of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction methods used by ATLAS, and more examples can be found in References [9–11]. But these five methods cover some of the most recent and readily available techniques offered to new physics analyses. They are listed here, but further details on their composition and differences are found throughout this chapter.

The first four working points are object-based, in that they include fully identified and calibrated objects in the vector sum. Loose $E_{\mathrm{T}}^{\mathrm{miss}}$, Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ and FJVT $E_{\mathrm{T}}^{\mathrm{miss}}$

take their names from the collection of jets used in the sum. Calo $E_{\mathrm{T}}^{\mathrm{miss}}$ is the fourth working point, and it is the only algorithm that includes calorimeter signals not associated with identified objects. The final working point named Track $E_{\mathrm{T}}^{\mathrm{miss}}$ does not used calibrated objects at all. Instead, it only uses the collection of ID tracks associated with the hard-scatter.

The genuine transverse momentum carried away by all undetected particles in an event is referred to as the True $E_{\mathrm{T}}^{\mathrm{miss}}$. This value can be calculated in simulated collisions, as the simulation may record all generator level particles which do not interact with the detector. It is obviously not available in data, though its expectation value can be approximated in some final states. At ATLAS the $Z \rightarrow ll$ final state, where l is either an electron or a muon, can be selected with very high signal-to-background ratios and the production of neutrinos only happen through very rare heavy-flavour decays in the hadronic recoil. Events in this channel are considered to have True $E_{\mathrm{T}}^{\mathrm{miss}} = 0$.

This chapter is organised as follows. Section 7.1 describes the basics of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction at ATLAS. The different working points are detailed along with their strengths and disadvantages. Additional observables associated with $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction are also introduced. Section 7.2 explains several methods for determining $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction performance from derived quantities such as the mean, width or tail of measured $E_{\mathrm{T}}^{\mathrm{miss}}$ distributions. The results and figures are taken from existing ATLAS studies [9, 10], which investigated $E_{\mathrm{T}}^{\mathrm{miss}}$ performance on a variety of real and simulated SM datasets containing either zero ($Z \rightarrow ll$) or non-zero ($W \rightarrow l\nu$, $t\bar{t}$, etc) genuine missing transverse momentum. These standard ATLAS $E_{\mathrm{T}}^{\mathrm{miss}}$ performance metrics are vital since they are used to judge the $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstructed by the neural networks in Chapter 10. The derivation of systematic uncertainties associated with $E_{\mathrm{T}}^{\mathrm{miss}}$ measurements is not included in this chapter. While a significant part of $E_{\mathrm{T}}^{\mathrm{miss}}$ analysis, no $E_{\mathrm{T}}^{\mathrm{miss}}$ systematic uncertainties were used directly in this project due to the complications of propagating them through a completed neural network, as discussed in Section 12.1.

7.1 $\boldsymbol{E}_{\mathrm{T}}^{\mathrm{miss}}$ Basics

7.1.1 Object-Based $\boldsymbol{E}_{\mathrm{T}}^{\mathrm{miss}}$

Most current forms of $E_{\mathrm{T}}^{\mathrm{miss}}$ are object-based, and they are characterised by two distinct contributions. The first contribution is from hard-event signals which are from identified and fully calibrated electrons, photons, muons and jets (hard-objects)[†]. Each physics object is reconstructed independently from different subsets of detector signals following the dedicated procedures described in Chapter 6. The second

[†]Hadronically decaying τ-leptons are not typically included in this process and are left as fully calibrated jets. Some configurations exist which do explicitly include them [224].

contribution to object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ is from soft-event signals. These are detector signals which, due to either kinematic thresholds, quality requirements or ambiguity in their source, failed to be attributed to an identified hard-object.

The missing transverse momentum vector $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ is calculated using the components of these two contributions along the x- and y-axes.

$$\underbrace{\underbrace{\mathbf{E}_{\mathrm{T}}^{\mathrm{miss},e}}_{-\sum_{\text{selected electrons}} \mathbf{p}_{\mathrm{T}}^{e}} \underbrace{\mathbf{E}_{\mathrm{T}}^{\mathrm{miss},\mu}}_{-\sum_{\text{selected muons}} \mathbf{p}_{\mathrm{T}}^{\mu}} \underbrace{\mathbf{E}_{\mathrm{T}}^{\mathrm{miss},\gamma}}_{-\sum_{\text{selected photons}} \mathbf{p}_{\mathrm{T}}^{\gamma}} \underbrace{\mathbf{E}_{\mathrm{T}}^{\mathrm{miss,jet}}}_{-\sum_{\text{selected jets}} \mathbf{p}_{\mathrm{T}}^{\mathrm{jet}}}}_{\text{hard-terms}} \underbrace{\mathbf{E}_{\mathrm{T}}^{\mathrm{miss,soft}}}_{-\sum_{\text{unused signals}} \mathbf{p}_{\mathrm{T}}^{\mathrm{soft}}}}_{\text{soft-term}} \tag{7.1}$$

The magnitude and azimuthal angle (ϕ^{miss}) of the vector $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ are calculated using its components $(E_x^{\mathrm{miss}}, E_y^{\mathrm{miss}})$ by:

$$\begin{aligned} E_{\mathrm{T}}^{\mathrm{miss}} &= \sqrt{(E_x^{\mathrm{miss}})^2 + (E_y^{\mathrm{miss}})^2} \\ \phi^{\mathrm{miss}} &= \arctan\left(E_y^{\mathrm{miss}} / E_x^{\mathrm{miss}}\right) \end{aligned} \tag{7.2}$$

By definition $E_{\mathrm{T}}^{\mathrm{miss}}$ is non-negative. In an experimental environment where not all relevant p_{T} from the hard-scatter interaction is incorporated into Equation 7.1, and the measured momenta from each contributing object is affected by the limited resolution of the detector, an observation bias towards non-vanishing values for $E_{\mathrm{T}}^{\mathrm{miss}}$ is introduced. This observation bias is most notable in final states without True $E_{\mathrm{T}}^{\mathrm{miss}}$.

An additional but also vital observable is the total transverse energy measured in the detector. This is labelled as ΣE_{T} and it quantifies the total event activity and may be interpreted as the hardness of the interaction. It provides a scale to measure $E_{\mathrm{T}}^{\mathrm{miss}}$ response and resolution, and is calculated using a scalar sum of the same transverse momenta terms, both hard and soft, that contribute to $E_{\mathrm{T}}^{\mathrm{miss}}$.

$$\Sigma E_{\mathrm{T}} = \underbrace{\sum_{\text{selected electrons}} p_{\mathrm{T}}^{e}}_{\Sigma E_{\mathrm{T}}^{e}} + \underbrace{\sum_{\text{selected muons}} p_{\mathrm{T}}^{\mu}}_{\Sigma E_{\mathrm{T}}^{\mu}} + \underbrace{\sum_{\text{selected photons}} p_{\mathrm{T}}^{\gamma}}_{\Sigma E_{\mathrm{T}}^{\gamma}} + \underbrace{\sum_{\text{selected jets}} p_{\mathrm{T}}^{\mathrm{jet}}}_{\Sigma E_{\mathrm{T}}^{\mathrm{jet}}} + \underbrace{\sum_{\text{unused signals}} p_{\mathrm{T}}^{\mathrm{track}}}_{\Sigma E_{\mathrm{T}}^{\mathrm{soft}}} \tag{7.3}$$

Hard-Terms

Dedicated identification and calibration procedures are carried out for each reconstructed hard physics object, translating detector signals into collections of fully corrected four-momenta. A subset of these objects is then selected for each hard-term in Equation 7.1. These subsets are constructed using quality and isolation criteria which attempts to reject fake or otherwise problematic signatures. In this project,

the selection of photons, electrons, and muons correspond to the baseline selection of objects defined in Chapter 6.2. The collection of jets however varies between the different $E_{\mathrm{T}}^{\mathrm{miss}}$ working points. Both the Tight and Calo $E_{\mathrm{T}}^{\mathrm{miss}}$ working points utilise the Tight collection of jets. The Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ and FJVT $E_{\mathrm{T}}^{\mathrm{miss}}$ take as inputs the Loose and FJVT collections of jets, respectively. Generally, the selection of hard-objects is refined to achieve optimal $E_{\mathrm{T}}^{\mathrm{miss}}$ performance within a given physics analysis, and therefore the selections used in this dissertation are just an example set of criteria which are typical but not universal.

In the calculation of $E_{\mathrm{T}}^{\mathrm{miss}}$ and ΣE_{T}, the contributing objects need to arise from mutually exclusive detector signals. However, each physics object is reconstructed and identified independently of one another. So, to prevent the multiple inclusions of the same signal, a process called signal ambiguity resolution is employed. Tracks and energy deposits in the ID are matched to reconstructed baseline objects, and a rejection mechanism based on a predefined order removes those which share signals with another higher priority object. Additional calibrations for only slightly overlapping objects are also carried out. This is a similar process to OR which was discussed in Chapter 6.3, but it has been specially developed to improve $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution.

The most commonly used order for the reconstruction sequence starts with muons, followed by electrons, then photons, and finally jets. Based on this sequence, all muons passing the baseline selection enter the $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction first. Each lower-priority reconstructed particles are fully rejected if they are found to share tracks or calorimeter deposits with a higher-priority object which has already entered the $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Jets are included last in the reconstruction, and they receive a less trivial treatment depending on what they overlap with.

Here the ambiguity resolution between overlapping electrons and jets is discussed, but the same procedure is applied to resolve photon and jet conflicts. If a jet is found close to a baseline electron, then a resolution takes place to determine how much of the jet's energy should be included in the $E_{\mathrm{T}}^{\mathrm{miss}}$ jet term. The main variable is the ratio of the electron energy to the jet energy, f_{overlap}, where both energies are calibrated at the EM scale. If a higher p_{T} jet is emitted close to an electron, then it may lead to $f_{\mathrm{overlap}} < 0.5$. In this case, the jet is included in the reconstruction but its p_{T} is scaled by f_{overlap}. Conversely, if $f_{\mathrm{overlap}} > 0.5$ then the jet is rejected, and any signals exclusively associated with it are left for the soft-term. Depending on how the soft-term is calculated, this could drastically undervalue the p_{T} of a real jet. More recent studies [9] have found that $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution may be improved by reducing the number of real jets which are wrongly assigned to the soft-term. This is done by allowing a jet to be treated as real, and to be placed in the jet term if both $f_{\mathrm{overlap}} < 1.0$ and the jet p_{T} is at least 20 GeV higher than the electron p_{T}. This latest technique is used in this dissertation.

Muons lose some energy when traversing the calorimeters, and a non-isolated muon

may overlap with other hard-objects including jets. A pileup jet containing an overlapping muon track will receive a higher JVT value, and thus may be mistakenly used in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. In addition, muons can themselves be mistaken as a jet due to significant energy loss in the calorimeters. This jet would then be found in close proximity to the ID track associated with the muon. If the jet was included into $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction it would lead to the double counting of the transverse momentum associated with this loss. This is because the energy loss is already corrected for in the fully calibrated muon p_{T}. Jets which are found close to muons are therefore rejected if at least one of the four following conditions is met.

- The ID track of the muon is ghost associated to the jet using the anti-k_t algorithm.
- The muon ID track represents more than 80% of the total of the transverse momentum of all jet tracks emerging from PV_0.
- The final calibrated jet p_{T} is less than twice the p_{T} of the ID track associated with the muon.
- The jet contains less than five ID tracks emerging from PV_0.

These jets are considered to be either from pileup interactions or from catastrophic muon energy loss. While these jets are rejected for $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction, the corresponding muons are accepted.

Muons may also radiate hard photons at small angles due to bremsstrahlung radiation. These are not typically reconstructed as photons due to the nearby ID track, as this would violate the baseline photon isolation requirement. Furthermore, due to the discrepancy between the energy measured by the calorimeter and the energy measured by the track, they would also fail electron reconstruction as well. Instead, these Final-State Radiation (FSR) photons are reconstructed as jets, but due to their proximity to the muon they fail the above muon-jet signal ambiguity resolution. So, jets meeting all the criteria consistent with the characteristics of a FSR photon are accepted for $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction [9], even if they overlap with a muon. They are re-calibrated at the correct energy scale reflecting their interpretation as a photon.

Soft-Terms

The soft-term in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction is a necessary inclusion as it represents additional detector signals that, due to kinematic thresholds or reconstruction inefficiencies, did not get associated with fully identified physics objects. Soft-signals also include those which were dropped during signal ambiguity resolution. These left-over signals still represent a significant fraction of the total transverse momentum of an event and contain contributions stemming from the hard-scatter, the underlying event and pileup interactions. Several algorithms designed to reconstruct and

calibrate the soft-term have been developed. In this dissertation, only two configurations are considered, but a broader comparison can be found in Reference [11]. The two algorithms are presented below, along with a description of the method and a motivation for their use in ATLAS.

The first configuration is the Calorimeter Soft-Term (CST). This reconstruction algorithm uses information from the calorimeters. While it does apply small corrections based on tracking information it does not attempt to differentiate between pileup and hard-scatter signals. The CST includes p_{T} contributions from energy deposits in the calorimeter which were not matched to, and did not overlap with, hard physics objects already used in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Noise suppression is applied to avoid the inclusion of fake calorimeter signals. This is achieved by only using cells belonging to topoclusters with positive energies after a calibration at the local cluster weighting (LCW) scale [225, 226]. Some tracks are included in the CST if they are matched to a soft topocluster but not a hard-object. For these matched signals an energy flow algorithm is used to determine which measurement to use. ID tracks with $p_{\mathrm{T}} > 0.4\,\mathrm{GeV}$ are used instead of the topocluster if their p_{T} resolution is expected to be superior than the calorimeter p_{T} resolution.

The second soft-term algorithm studied in this dissertation is the Track Soft-Term (TST). This term is reconstructed purely from ID tracks with $p_{\mathrm{T}} > 0.4\,\mathrm{GeV}$ associated with the hard scatter, but not matched to any hard physics objects. Tracks are also excluded if they do not leave any hits in the pixel detector or less than 6 hits in the SCT, which ensures that the p_{T} measurements of the selected tracks are reliable. Tracks are matched to the PV_0 by applying the following restrictions based on their impact parameters: $|d_0/\sigma_{d_0}| < 2$ and $|z_0 \sin\theta| < 3\,\mathrm{mm}$. Tracks are also restricted by the coverage of the ID tracking volume of $|\eta| < 2.5$. To prevent the inclusion of tracks stemming from hard-objects, all of the following are excluded from the TST:

- Tracks found within $\Delta R = 0.05$ of an electron or a photon.
- Tracks matched to jets using the ghost association technique [153].
- ID tracks associated with identified muons.
- Isolated tracks with $p_{\mathrm{T}} > 120\,\mathrm{GeV}$ which have an estimated relative resolution on their p_{T} larger than 40%.
- Isolated tracks with $p_{\mathrm{T}} > 120\,\mathrm{GeV}$ which have no associated energy deposit in the calorimeter with a p_{T} larger than 65% of the track p_{T}.

The last two requirements are to reject mismeasured tracks, while still accepting muons not in the coverage of the MS. No calorimeter topoclusters are included at all in the TST.

All terms in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction are affected by pileup, but the most affected are the jet and CST term since their constituents are spread over larger regions in the

calorimeters. The calorimeters have much slower reset times and are therefore more affected by out-of-time pileup. Furthermore, the CST does not suppress signals from in-time pileup interactions and significant deterioration in the CST resolution is observed as the average number of these interactions increases [11]. While the CST was the standard configuration for the soft-term in most ATLAS analyses at 7 and 8 TeV during Run 1, the recommended method for analyses performed on Run 2 data, which has significantly more interactions per bunch crossing, is the TST. Despite the fact that the TST misses all contributions from soft neutral particles, it performs excellent vertex matching for the soft-term and is not affected by in-time and out-of-time pileup.

In this dissertation, the TST was used for Loose $E_{\mathrm{T}}^{\mathrm{miss}}$, Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ and FJVT $E_{\mathrm{T}}^{\mathrm{miss}}$. The only working point which used the CST was Calo $E_{\mathrm{T}}^{\mathrm{miss}}$.

7.1.2 Track only $\boldsymbol{E}_{\mathbf{T}}^{\mathbf{miss}}$

An extension of the philosophy and method of the TST is the reconstruction of $E_{\mathrm{T}}^{\mathrm{miss}}$ using only ID tracks. This method of reconstruction does not include any object identification or selection and thus is dissimilar to the other four object oriented working points. Track $E_{\mathrm{T}}^{\mathrm{miss}}$ performance has almost no pileup dependence, however it does suffer due to the lack of inclusion of neutral particles which do not leave any tracks in the ID. This is particularly noticeable in event topologies with numerous or highly energetic jets and photons. Furthermore, the acceptance of signals for Track $E_{\mathrm{T}}^{\mathrm{miss}}$ is limited to only the ID coverage of $|\eta| < 2.5$, less than the calorimeter coverage which extends up to $|\eta| < 4.9$. Track $E_{\mathrm{T}}^{\mathrm{miss}}$ is calculated by the negative vectorial sum of $\mathbf{p}_{\mathrm{T}}$ of all ID tracks using the same selection requirements as the TST without the hard-object overlap removal. This includes the removal of tracks with either poor momentum resolution or without corresponding calorimeter deposits using the same criteria as described in 7.1.

7.1.3 $\boldsymbol{E}_{\mathbf{T}}^{\mathbf{miss}}$ Significance

Mismeasurements in $E_{\mathrm{T}}^{\mathrm{miss}}$ can arise due to detector inefficiencies, incomplete detector coverage, and interacting particles which are incorrectly reconstructed or fail to be reconstructed all-together. These sources of fake $E_{\mathrm{T}}^{\mathrm{miss}}$ complicate the conclusion of the existence or non-existence of undetectable particles based on the observed $E_{\mathrm{T}}^{\mathrm{miss}}$ alone. Therefore, another variable called the $E_{\mathrm{T}}^{\mathrm{miss}}$ significance $\mathcal{S}$ can be used instead. On an event by event basis, $\mathcal{S}$ approximates the p-value using a log-likelihood ratio that the measured $E_{\mathrm{T}}^{\mathrm{miss}}$ is consistent with the null hypothesis. In this case the null hypothesis is that the True $E_{\mathrm{T}}^{\mathrm{miss}} = 0$, and the reconstructed $E_{\mathrm{T}}^{\mathrm{miss}} > 0$ is consistent with detector resolution and particle identification efficiencies. A large value of $\mathcal{S}$ would indicate that the observed $E_{\mathrm{T}}^{\mathrm{miss}}$ cannot be explained by limitations in the detector and suggests that the event does in fact contain non-interacting objects.

ATLAS and CMS have previously defined event-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance $\mathcal{S}_E$ as either:

$$\mathcal{S}_E = \frac{E_{\mathrm{T}}^{\mathrm{miss}}}{\sqrt{H_{\mathrm{T}}}} \text{ or } \mathcal{S}_E = \frac{E_{\mathrm{T}}^{\mathrm{miss}}}{\sqrt{\Sigma E_{\mathrm{T}}}}, \tag{7.4}$$

where H_{T} is the scalar sum of the transverse momenta of all hard-objects, and is equivalent to ΣE_{T} from Equation 7.3 without the inclusion of the soft-term. These likelihoods are based on the assumption that $E_{\mathrm{T}}^{\mathrm{miss}}$ is derived from calorimeter signals only.

To better reflect current reconstruction methods, a more complex and object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance variable was developed, $\mathcal{S}_O$ [224]. This variable takes into consideration the directional correlations between measurements, the expected resolutions, and likelihood of mismeasurement of all the objects that enter $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. For each of the hard-objects, one can define a transverse momentum resolution $\sigma_{p_{\mathrm{T}}}$, and an angular resolution in the transverse plane σ_ϕ. These measurements are taken to be uncorrelated. The specific values used for these resolutions depend on the object type, p_{T}, and the quality of the signal recorded by the detector. They are derived from dedicated resolution studies for electrons and photons [227], muons [206], and jets [216, 226, 228]. The resolution of the soft-term is taken from a study of $Z \to \mu\mu$ events [224], and has a set value of $\sigma_{\mathrm{soft}} = 8.9\,\mathrm{GeV}$ regardless of p_{T} or azimuthal angle.

The object-based significance is defined by the log ratio:

$$\mathcal{S}_O^2 = 2\ln\left(\frac{\max_{\text{True } E_{\mathrm{T}}^{\mathrm{miss}}>0} \mathcal{L}(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|\text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}})}{\max_{\text{True } E_{\mathrm{T}}^{\mathrm{miss}}=0} \mathcal{L}(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|\text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}})}\right). \tag{7.5}$$

The likelihood functions depend on the multiplicities, types, and kinematics of the objects in the event that enter $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction, represented by i in the following equations. They are calculated on an event by event basis using the following assumptions:

- The measurement of each reconstructed object i is independent from others.
- The probability distribution of measuring $\mathbf{p}_{\mathrm{T}}{}^i$ given the true object transverse momentum $\mathbf{q}_{\mathrm{T}}{}^i$ is defined by a two dimensional Gaussian with mean $\mathbf{q}_{\mathrm{T}}{}^i$ and covariance matrix $\mathbf{V}^i$.
- All relevant objects are accounted for, such that $\sum_i \mathbf{q}_{\mathrm{T}}{}^i = \text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$

Under these assumptions the likelihood function takes the form of a two dimensional Gaussian and the log-likelihood ratio is reduced to a chi square χ^2 variable with two degrees of freedom:

$$\mathcal{S}_O^2 = 2\ln\left(\frac{\mathcal{L}(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}})}{\mathcal{L}(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|\mathbf{0})}\right) = (\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}})^{\intercal}\left(\sum_i \mathbf{V}^i\right)^{-1}(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}). \tag{7.6}$$

From Equation 7.6, it is visible how the resolutions of the included physics objects, through their covariance matrices, influence the overall level of significance to an $E_{\mathrm{T}}^{\mathrm{miss}}$ estimate. For each hard-object, the matrices are defined using an orthogonal coordinate system whereby one axis is aligned with the measured ${\mathbf{p}_{\mathrm{T}}}^{i}$:

$$\mathbf{V}^i = \begin{pmatrix} \sigma^2_{{p_{\mathrm{T}}}^i} & 0 \\ 0 & {p_{\mathrm{T}}}^{i\,2}\sigma^2_{\phi^i} \end{pmatrix}. \tag{7.7}$$

The covariance matrix of the soft-term is also added into Equation 7.6. It is defined as

$$\mathbf{V}^{\mathrm{soft}} = \begin{pmatrix} \sigma_{\mathrm{soft}} & 0 \\ 0 & \sigma^2_{\mathrm{soft}} \end{pmatrix}. \tag{7.8}$$

The covaraince matrices are summed together in the standard ATLAS $x-y$ coordinate system, and therefore must be individually transformed:

$$\mathbf{V}_{xy} = \sum_i R^{-1}\left(\phi^i\right)\mathbf{V}^i R\left(\phi^i\right) + \mathbf{V}^{\mathrm{soft}}, \tag{7.9}$$

where $R\left(\phi^i\right)$ is the two dimensional rotation matrix in the azimuthal direction for each object. This total covariance matrix is rotated once more to be described by the directions parallel and perpendicular to $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$:

$$\mathbf{V}_{LT} = R^{-1}\left(\phi^{\mathrm{miss}}\right)\mathbf{V}_{xy}R\left(\phi^{\mathrm{miss}}\right) = \begin{pmatrix} \sigma^2_{\mathrm{L}} & \rho_{\mathrm{LT}}\sigma_{\mathrm{L}}\sigma_{\mathrm{T}} \\ \rho_{\mathrm{LT}}\sigma_{\mathrm{L}}\sigma_{\mathrm{T}} & \sigma^2_{\mathrm{T}} \end{pmatrix}, \tag{7.10}$$

where σ^2_{L} and σ^2_{T} are the total variances in the longitudinal and transverse directions of $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ respectively, and ρ_{LT} is the correlation factor between the two directions.

Using this basis, the object based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance can be written as:

$$\mathcal{S}^2_O = \left(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}\right)^{\intercal}\left(\mathbf{V}_{LT}\right)^{-1}\left(\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}\right) = \frac{{E_{\mathrm{T}}^{\mathrm{miss}}}^2}{\sigma^2_{\mathrm{L}}\left(1-\rho^2_{\mathrm{LT}}\right)}. \tag{7.11}$$

It was found that $\mathcal{S}_O$ was a superior discriminating variable than either $E_{\mathrm{T}}^{\mathrm{miss}}$ or $\mathcal{S}_E$ (using the Loose working point and ΣE_{T} in the denominator) when it came to separating simulated $Z \to ee$ and $ZZ \to ee\nu\nu$ events, especially in the presence of jets, as shown in Figure 7.1.

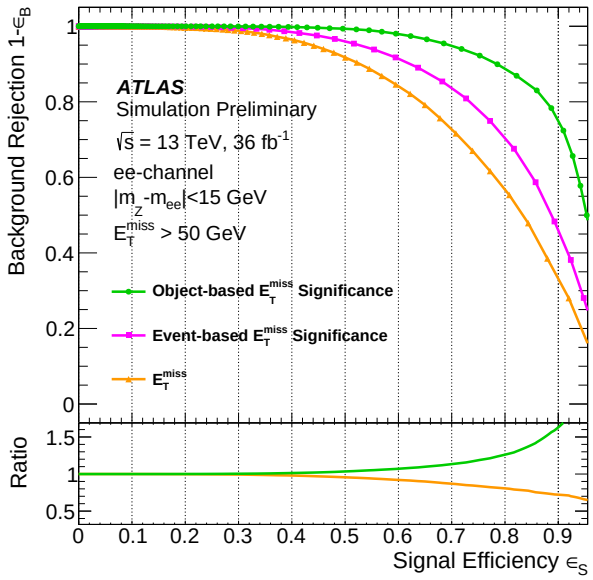


Figure 7.1: A receiver operating characteristic curve showing the background ($Z \to ee$) rejection versus signal ($ZZ \to ee\nu\nu$) efficiency in simulated samples [224]. The performance is shown using $E_{\mathrm{T}}^{\mathrm{miss}}$, event-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance ($\mathcal{S}_E$), and object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance ($\mathcal{S}_O$) as discriminants. The lower panel shows the ratio of the other curves against $\mathcal{S}_E$.

7.2 Performance of $E_{\mathrm{T}}^{\mathrm{miss}}$

The performance of an $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm is characterised by several metrics. Most notably these are the response of the measurement and its resolution, but a more detailed analysis is required to fully convey the strengths and weaknesses of a particular method. Response and resolution functions are characterised by a high level of complexity due to the composite nature of the observable, taking contributions from many objects each with different p_{T} resolutions. Even between events with the same final state, $E_{\mathrm{T}}^{\mathrm{miss}}$ composition can fluctuate significantly. This is due to the many varying sources of fake $E_{\mathrm{T}}^{\mathrm{miss}}$. Accurate missing transverse momentum measurements can only be achieved in a perfectly hermetic detector which is one that records the kinematics of every known interacting particle with a full 4π solid angle of coverage. While ATLAS is considered an example of a hermetic detector, in practice the limited detector coverage of $|\eta| < 4.9$, as well as gaps in the calorimeters, restrict the set of particles which can contribute to $E_{\mathrm{T}}^{\mathrm{miss}}$. In addition, the existence of irreducible signal fluctuations in the different sub-detectors degrade $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Both in-time and out-of-time pileup interactions obscure the hard-scatter and pure separation between pileup and hard-scatter signals is impossible, and pileup contributions inevitably get included in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. The need to suppress pileup signals in a detector where coverage and reconstruction efficiencies are inconsistent and dependent on the type of particle lead to varying $E_{\mathrm{T}}^{\mathrm{miss}}$ performance. Therefore, $E_{\mathrm{T}}^{\mathrm{miss}}$ response and resolution must be understood within

the context of a given final state event topology, scale, overall event activity and the amount of pileup.

Another obstacle when determining the accuracy of an $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm is that it requires some comparison to a known or expected value. The performance of $E_{\mathrm{T}}^{\mathrm{miss}}$ is therefore evaluated in two event types.

The first of those types are events likely to contain genuine missing transverse momentum. Here performance is calculated using MC simulations as True $E_{\mathrm{T}}^{\mathrm{miss}}$ is readily available. This does imply that the understanding of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction is limited by how well the simulation matches the real interactions, so validation and systematic uncertainties in $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution can be derived from MC-to-data comparisons. The uncertainties associated with the detector simulation can be propagated to the overall $E_{\mathrm{T}}^{\mathrm{miss}}$ uncertainty for a given event.

The second type of events are those unlikely to produce genuine missing transverse momentum. Here $E_{\mathrm{T}}^{\mathrm{miss}}$ performance can be investigated directly in data, but only if the final states heavily favour these processes. As mentioned above, at ATLAS this process is the decay of a Z boson into a muon or electron pair. Any measured value of $E_{\mathrm{T}}^{\mathrm{miss}}$ on such events can be considered as arising from fake sources only. Since the reconstructed final state can be produced by other physics processes, data-to-MC comparisons are still used to validate the signal region composition. Each contributing process is accounted for in MC and are scaled according to their cross-section. This also allows the identification of potential mismodelling.

Different methods for calculating $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution and response are employed depending on whether or not the $E_{\mathrm{T}}^{\mathrm{miss}}$ signature is considered to be true or fake. Each of the $E_{\mathrm{T}}^{\mathrm{miss}}$ working points offer different methods to improve some combination of the $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution, scale and stability against pileup. As was the case with the choice of criteria for OS, the particular choice of $E_{\mathrm{T}}^{\mathrm{miss}}$ working point used for a given analysis strongly depends on its specific performance requirements.

7.2.1 Response

In the context of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction, the response is defined as how the observed $E_{\mathrm{T}}^{\mathrm{miss}}$ deviates from an expectation value for a given final state. This deviation is usually plotted as a function of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ or another variable indicative of the hard-scatter activity and sets the scale for the observed $E_{\mathrm{T}}^{\mathrm{miss}}$. If this relationship is independent then the $E_{\mathrm{T}}^{\mathrm{miss}}$ response is deemed to be linear. A linear response in $E_{\mathrm{T}}^{\mathrm{miss}}$ with a constant deviation is said to have a bias. Final states which are unlikely to have genuine missing transverse momentum are expected to show a non-linear $E_{\mathrm{T}}^{\mathrm{miss}}$ response especially at low event activity. This is predominantly due to the observation bias that affects all methods of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. In final states with genuine missing transverse momentum, the response only becomes linear once True $E_{\mathrm{T}}^{\mathrm{miss}}$ exceeds this observation bias.

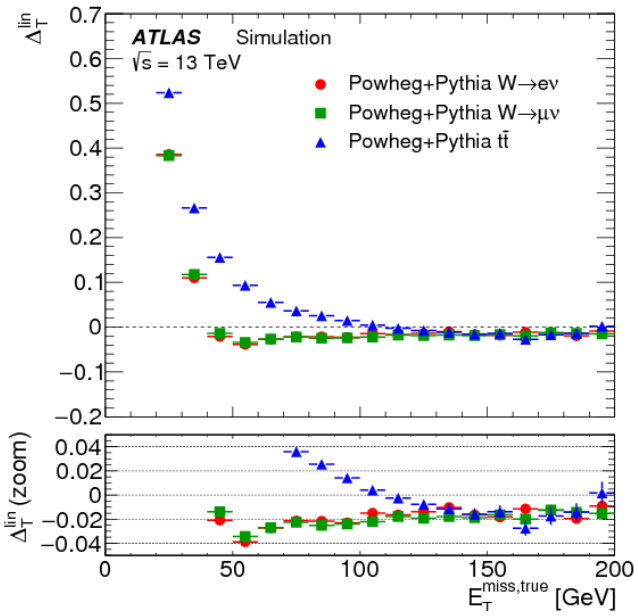


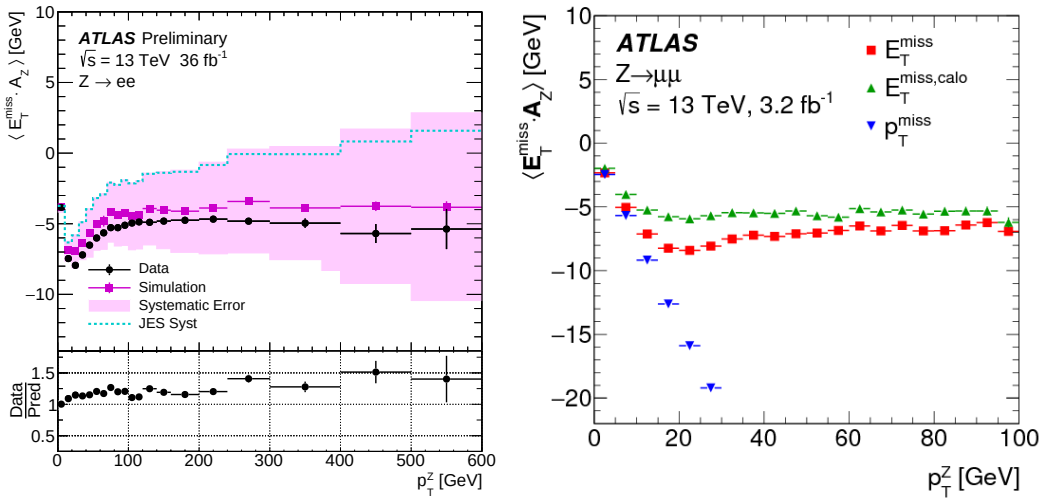
Figure 7.2: The deviation of the $E_{\mathrm{T}}^{\mathrm{miss}}$ response from linearity, measured as a function of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ in $W \to e\nu$, $W \to \mu\nu$, and $t\bar{t}$ final states in MC simulations [10]. The lower plot shows a zoomed-in view.

For final states with genuine missing transverse momentum, response is determined in MC simulated events. This is done by evaluating the relative deviation Δ_T^{lin} of the reconstructed $E_{\mathrm{T}}^{\mathrm{miss}}$ as a function of True $E_{\mathrm{T}}^{\mathrm{miss}}$. This is referred to as the linearity of response. This value is expected to be zero if the $E_{\mathrm{T}}^{\mathrm{miss}}$ is reconstructed at the correct scale.

$$\Delta_T^{\mathrm{lin}}(\text{True } E_{\mathrm{T}}^{\mathrm{miss}}) = \frac{E_{\mathrm{T}}^{\mathrm{miss}} - \text{True } E_{\mathrm{T}}^{\mathrm{miss}}}{\text{True } E_{\mathrm{T}}^{\mathrm{miss}}} \tag{7.12}$$

The linearity of response was investigated in Reference [10] and a plot comparing how its behaviour changed with respect to different event topologies is shown in Figure 7.2. The $E_{\mathrm{T}}^{\mathrm{miss}}$ method used in the creation of this plot is equivalent to the Loose working point. The observed $\Delta_T^{\mathrm{lin}} > 0$ at low True $E_{\mathrm{T}}^{\mathrm{miss}}$ values is due to the aforementioned observation bias and indicates that the measured $E_{\mathrm{T}}^{\mathrm{miss}}$ is still within the scale of its resolution. As mentioned, jet p_{T} resolution is particularly sensitive to pileup. So due to the higher number of jets, $E_{\mathrm{T}}^{\mathrm{miss}}$ response and resolution is observed to be worse on the $t\bar{t}$ sample than the W-boson samples. The negative values of Δ_T^{lin} for the W samples indicate an underestimation of the hadronic recoil. At high True $E_{\mathrm{T}}^{\mathrm{miss}}$ values the $E_{\mathrm{T}}^{\mathrm{miss}}$ response is directly proportional and is approximately 2% too small.

In events with the $Z \to ll$ final state, the transverse momentum of the Z boson (p_{T}^Z) can be seen as an indicator of the hardness of the interaction and may be used as for investigating the $E_{\mathrm{T}}^{\mathrm{miss}}$ response. The direction of the Z boson defines an axis $\mathbf{A}_Z$ in the transverse plane of the collision which is reconstructed from the measured



(a) (b)

Figure 7.3: The average value $\langle \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} \cdot \mathbf{A}_Z \rangle$ is shown versus p_{T}^Z in a (a) $Z \to ee$ and a (b) $Z \to \mu\mu$ final state. For the $Z \to ee$ sample some small non-Z backgrounds were included in the simulation [9]. Detector level uncertainties are indicated by the pink band. The largest source of systematic uncertainty comes from the jet energy scale (JES). Three different $E_{\mathrm{T}}^{\mathrm{miss}}$ working points are compared for the $Z \to \mu\mu$ final state [10], corresponding to Loose (red), Calo (green) and Track (blue) $E_{\mathrm{T}}^{\mathrm{miss}}$.

kinematics of the two leptons. The variable of interest is $\mathcal{P}_{||}^Z$ which is equal to the magnitude of the component of $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ parallel to $\mathbf{A}_Z$.

$$\mathbf{A}_Z = \frac{\mathbf{p}_{\mathrm{T}}^{l^+} + \mathbf{p}_{\mathrm{T}}^{l^-}}{|\mathbf{p}_{\mathrm{T}}^{l^+} + \mathbf{p}_{\mathrm{T}}^{l^-}|} = \frac{\mathbf{p}_{\mathrm{T}}^Z}{p_{\mathrm{T}}^Z} \tag{7.13}$$

$$\mathcal{P}_{||}^Z = \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} \cdot \mathbf{A}_Z \tag{7.14}$$

Unlike the deviation from linearity mentioned above, $\mathcal{P}_{||}^Z$ can be determined for both data and MC simulation, thus providing an important tool for the validation of the $E_{\mathrm{T}}^{\mathrm{miss}}$ response. For any balanced interaction the expectation value of this projection $\langle \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} \cdot \mathbf{A}_Z \rangle = 0$. Any deviation is sensitive to any limitation in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction both in terms of response and resolution. However, since the resolution of the lepton p_{T} is relatively high, this value is particularly dependent on the contribution from the hadronic recoil against the Z boson. At the low p_{T} scale the hadronic recoil is represented in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction in the soft-term.

Response is shown in Figure 7.3(a) for both a real and simulated $Z \to ee$ sample using the Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ working point. The steep decrease of $\langle \mathcal{P}_{||}^Z \rangle$ with increasing p_{T}^Z indicates an inherent underestimation of the hadronic recoil. This is because the hadronic recoil enters $E_{\mathrm{T}}^{\mathrm{miss}}$ via the TST which does not capture neutral energy. For $p_{\mathrm{T}}^Z > 20\,\mathrm{GeV}$ it recovers towards zero as more of the hadronic recoil is reconstructed

as a fully calibrated jet. Figure 7.3(b) shows a comparison of the response determined by the Loose, Calo and Track $E_{\mathrm{T}}^{\mathrm{miss}}$ working points on an inclusive $Z \rightarrow \mu\mu$ sample. Here the Calo working point has a much better response, indicating a better representation of the hadronic recoil. Also visible is the significant degradation due to the exclusion of hard-terms, especially jets, associated with the Track working point.

7.2.2 Resolution

For $Z \rightarrow ll$ events the measured E_x^{miss} and E_y^{miss} distributions are expected to be independent and approximately Gaussian [229]. Deviations arise from noise and events with particularly large ΣE_{T}. These distributions have non-Gaussian tails which are particularly noticeable for the pileup suppressing algorithms. The resolution of $E_{\mathrm{T}}^{\mathrm{miss}}$ is therefore defined as the root mean square (RMS) of the combined E_x^{miss} and E_y^{miss} distributions. The RMS includes important information which is sensitive to the size of the tails which would be lost if a Gaussian fit over the core was used instead.

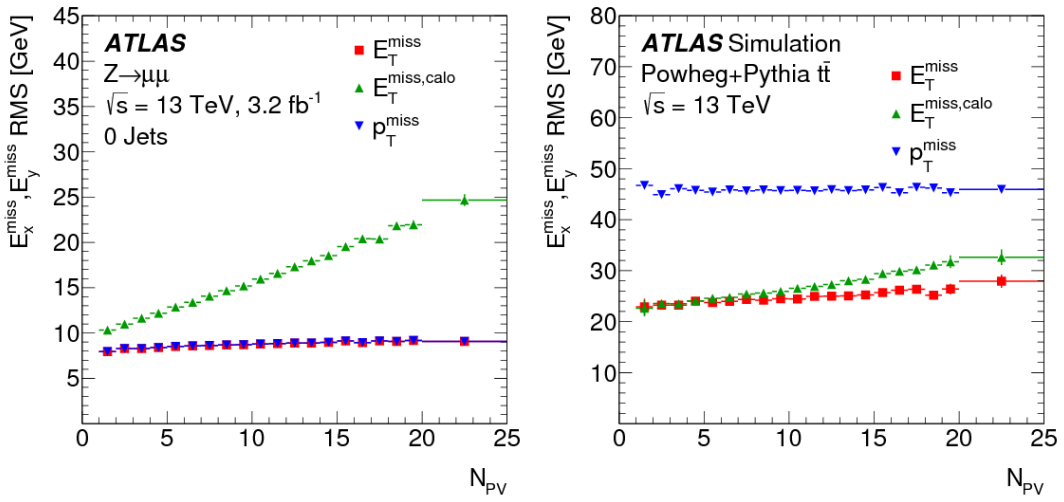
For processes with non-zero True $E_{\mathrm{T}}^{\mathrm{miss}}$ the resolution can only be calculated from simulation. Here the True $E_{\mathrm{T}}^{\mathrm{miss}}$ components are first subtracted from their respective measured quantities. Therefore, $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution for all final state events is defined by the root mean squared deviation or the root mean squared error (RMSE) of the combined component distributions as shown by Equation 7.15.

$$\mathrm{RMSE} = \begin{cases} \mathrm{RMS}(E_{x(y)}^{\mathrm{miss}}) & \text{True } E_T^{\mathrm{miss}} = 0, \text{ MC and data } (Z \rightarrow ll) \\ \mathrm{RMS}(E_{x(y)}^{\mathrm{miss}} - \text{True } E_{x(y)}^{\mathrm{miss}}) & \text{True } E_T^{\mathrm{miss}} > 0, \text{ MC only} \end{cases} \tag{7.15}$$

This metric does not capture all of the artefacts driving the fluctuations in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Biases between $E_{\mathrm{T}}^{\mathrm{miss}}$ terms or specific behaviours of outliers are not recorded in $\mathrm{RMSE}^{\mathrm{miss}}$ which is why further metrics such as the tail fraction is used in Section 7.2.4. Notwithstanding these limitations, this metric is still an appropriate general measure of how well $E_{\mathrm{T}}^{\mathrm{miss}}$ represents True $E_{\mathrm{T}}^{\mathrm{miss}}$.

One of the main tests of an $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm is the stability of its performance with an increase in pileup. This is performed by investigating the relationship of the $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution as a function of the number of reconstructed vertices N_{PV} or the average number of interactions per proton bunch crossing μ. Both variables are used since N_{PV} falls off its linear relationship with μ due to vertex merging [230] at higher instantaneous luminosities. Pileup sensitivity is a critical feature of $E_{\mathrm{T}}^{\mathrm{miss}}$ performance as the LHC increases its instantaneous luminosity with each Run.

Figure 7.4(a) shows the relationship between $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution and N_{PV} for the Loose, Calo and Track $E_{\mathrm{T}}^{\mathrm{miss}}$ working points on an exclusive $Z \rightarrow \mu\mu$ sample in data with

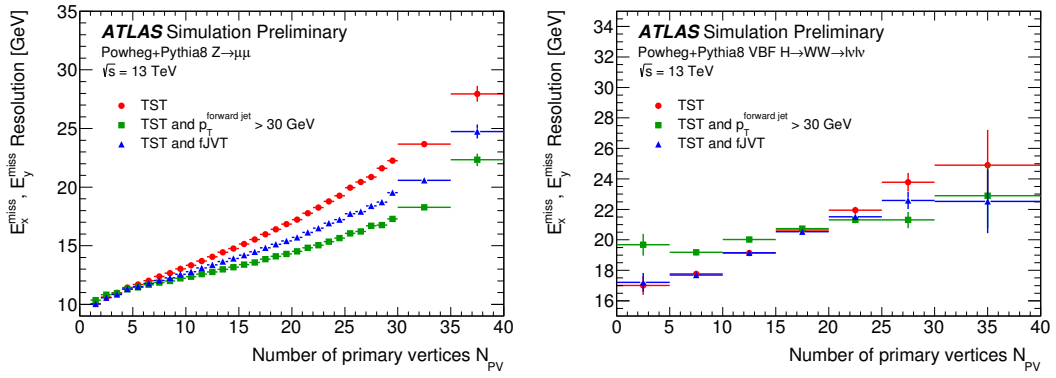


(a) (b)

Figure 7.4: The resolutions of three $E_{\mathrm{T}}^{\mathrm{miss}}$ working points as a function of N_{PV} [9]. The working points are equivalent to the Loose (red), Calo (green) and Track (blue) working points presented in this dissertation. The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is plotted as a function of pileup activity measured in terms of N_{PV} for (a) an exclusive $Z \to \mu\mu$ sample with no hard jets and (b) a simulated $t\bar{t}$ sample [10].

zero jets [10]. The Calo working point is the most affected by pileup and in the absence of jets has the worst resolution of the three across the full pileup range. Since most of the momentum in the event is carried by the muons, the Loose and Track $E_{\mathrm{T}}^{\mathrm{miss}}$ working points are almost identical and are both insensitive to pileup. Figure 7.4(b) compares the same relationships on a simulated $t\bar{t}$ sample which does contain jets. In samples with high jet multiplicities and genuine missing transverse momentum, Track $E_{\mathrm{T}}^{\mathrm{miss}}$ performs significantly worse than both Calo and Loose $E_{\mathrm{T}}^{\mathrm{miss}}$, despite still being relatively independent of N_{PV}. Despite dedicated corrections to suppress pileup contributions in the jet response, irreducible fluctuations in the calorimeters from pileup still lead to the degradation of the jet energy resolution. This results in to poorer resolutions in jet p_{T} measurements and worse $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions for both the Calo and the Loose working points. The Calo is doubly affected due to its increased contribution of soft calorimeter signals which cannot be confidently pileup suppressed.

Figure 7.5 compares the resolution as a function of N_{PV} for the working points which only differ by their jet selections. This is studied in events with zero True $E_{\mathrm{T}}^{\mathrm{miss}}$ in $Z \to \mu\mu$ simulation in Figure 7.5(a) and in simulated vector boson fusion (VBF) events which have high forward-jet multiplicities in Figure 7.5(b). In both topologies, increasing the forward-jet p_{T} threshold to 30 GeV reduces the pileup dependency of the resolution. This is because this region of phase space typically contains more pileup jets than hard-scatter jets. This does still result in the rejection of some hard-scatter jets, the effects of which are seen in the decrease in resolution for the VBF sample across the lower half of the pileup range. The FJVT working point



(a) (b)

Figure 7.5: The resolutions of three $E_{\mathrm{T}}^{\mathrm{miss}}$ working points as a function of N_{PV} [9]. The working points are equivalent to the Loose (red), Tight (green) and FJVT (blue) working points presented in this dissertation. The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is plotted as a function of pileup activity measured in terms of N_{PV} for (a) a simulated $Z \rightarrow \mu\mu$ sample and (b) a simulated VBF $H \rightarrow WW$ sample.

seems to offer a middle ground for pileup resilience without sacrificing accuracy in topologies with many forward-jets.

The resolution of $E_{\mathrm{T}}^{\mathrm{miss}}$ can be shown as a function of the total event activity defined by ΣE_{T}. An example of this relationship is shown in Figure 7.6, which was performed on an inclusive $Z \rightarrow \mu\mu$ final state using the Loose, Calo and Track working points. The Loose definition of ΣE_{T} is plotted along the x-axis for each working point to allow proper comparisons. In the low ΣE_{T} region the events have few, if any, jets leading to Calo yielding a poorer resolution than Loose and Track, which have near identical performance. In the high ΣE_{T} region, dominated by higher jet multiplicity, Track $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is degraded relative to Loose due to the incomplete representation of jets.

7.2.3 Angular Resolution

Angular resolution is another metric used to evaluate $E_{\mathrm{T}}^{\mathrm{miss}}$ performance. It is calculated from the RMS of the distribution containing the difference between the azimuthal angles of the measured $E_{\mathrm{T}}^{\mathrm{miss}}$ vector and the True $E_{\mathrm{T}}^{\mathrm{miss}}$. It can only be calculated in simulation and on events with non-zero genuine missing transverse momentum. Angular resolution is expected to gradually increase with True $E_{\mathrm{T}}^{\mathrm{miss}}$ for most algorithms since the direction of a vector becomes easier to gauge as its magnitude increases. Measuring ϕ^{miss} with high degrees of accuracy is particularly important for the reconstruction of kinematic observables such as the transverse mass [5] which is used for many different studies including measuring the W boson mass. It is also needed to calculate the mass of the Higgs boson in $H \rightarrow \tau\tau$ events [231].

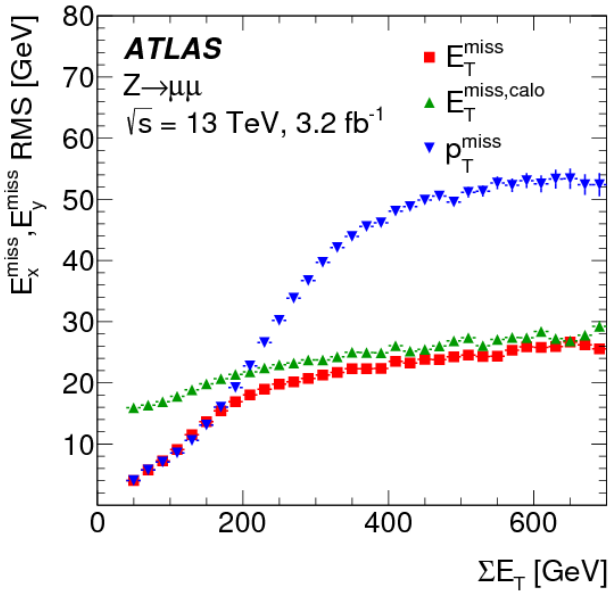


Figure 7.6: A comparison of three $E_{\mathrm{T}}^{\mathrm{miss}}$ definitions which are equivalent to the Loose (red), Calo (green) and Track (blue) working points presented in this dissertation. The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is plotted as a function of the total event activity defined by ΣE_{T} using the Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ configuration [10].

7.2.4 Distribution Tails

Unusually large or unexpected $E_{\mathrm{T}}^{\mathrm{miss}}$ values are potentially indicators of a new undetectable particles. Since such phenomena are key in searches for new physics it is useful to understand the likelihood of such a measurement occurring due to poor reconstruction. As mentioned above, the $E_{\mathrm{T}}^{\mathrm{miss}}$ distributions produce shapes with non-Gaussian tails. These tails arise from a combination of object selection inefficiencies and the individual p_{T} resolutions of constituent terms being themselves non-Gaussian. Even for a well-defined final state, event by event fluctuations in terms of which hard-objects and soft-signals enter the $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction can potentially lead to deviations from the normally distributed E_x^{miss} and E_y^{miss}. To negate this, the resolution of $E_{\mathrm{T}}^{\mathrm{miss}}$ has already been defined as the RMSE of these distributions rather than from a Gaussian fit. However, to further capture the extent of the tail, another metric can be used, f_{tail}. This variable is equal to the fraction of events in a sample where $|\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} - \text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|$ is greater than some threshold. Working points can be compared by the rate at which f_{tail} decreases as the threshold is increased. A faster decrease indicates less events presiding in the tail, which implies that a greater fraction was reconstructed with $E_{\mathrm{T}}^{\mathrm{miss}}$ closer to truth.

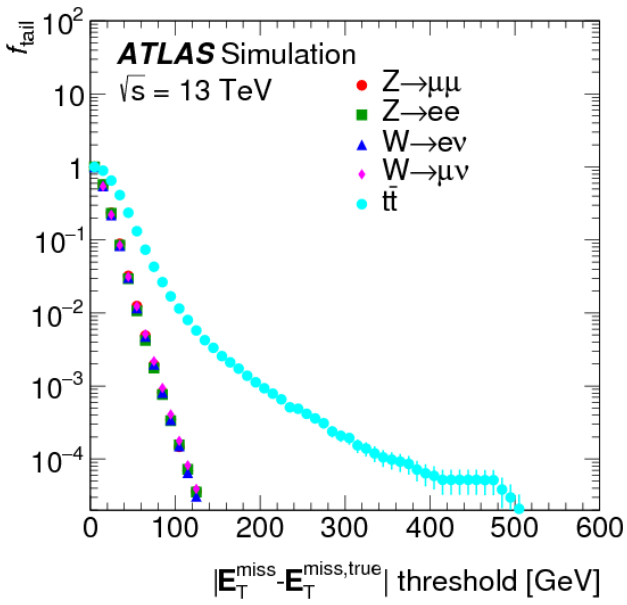


Figure 7.7: The tail fraction using a threshold applied to $|\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} - \text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|$, the euclidean distance between the reconstructed and true missing transverse momentum [10]. In this plot the Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ working point was used.

Figure 7.7 shows the tail distribution of several simulated samples using the same Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ working point. The tail distributions of the single boson decays are indifferent to the lepton flavour and boson type. Conversely the tail distribution for $t\bar{t}$ samples is considerably larger, suggesting that this difference is due to the much higher jet multiplicities and ΣE_{T} that is present in $t\bar{t}$ events, which are therefore more sensitive to pileup.

Chapter 8

Samples and Preselection

The practical aspects of this project involved the training of a deep neural network (Chapter 9), the determination of its performance using standard ATLAS $E_{\mathrm{T}}^{\mathrm{miss}}$ techniques (Chapter 10), and its application in a SUSY signal search (Chapter 11). Due to the limited number of available samples, many were used in more than one of these sections. So, all data and MC samples used throughout this dissertation are listed together in this chapter. They are then referenced by the following three chapters, indicating where they were used.

The three types of datasets include real data from pp collisions within ATLAS, MC simulated events from SM processes, and MC simulated events from SUSY processes. Network training took place exclusively on simulated datasets for reasons that are explained in Section 9.1, and not on the SUSY samples since they contained too few events to make any notable impact.

Standard supervised learning techniques require that the evaluation of finalised models is strictly conducted on events that were not seen during the learning phase. This required the categorisation of every simulated SM event into one of two orthogonal classes. These were named the Learning class* and the Evaluation class. The latter of which was used in both the SUSY search and the performance determination. Many different SM processes were used in this dissertation, and while some were dedicated entirely to a single class, others were manually partitioned. Table 8.1 shows to which class or classes each process was allocated, but the motivations and details of the splits are presented in Section 9.3.

8.1 Data

This analysis uses pp collision data captured by ATLAS and delivered by the LHC at $\sqrt{s} = 13\,\mathrm{TeV}$. The analysed data satisfies the standard quality selection criteria at ATLAS. It was recorded while the LHC declared stable beam conditions and all components of the ATLAS detector, including its magnets, were functioning normally. The data was collected by ATLAS in 2017 and has a total integrated luminosity of

*Which was further partitioned into the various training and testing sets.

$43\,\text{fb}^{-1}$. This provided sufficient statistics and a wide enough range in the number of interactions per crossing for the purposes of this work.

8.2 Monte Carlo Samples

The production of ATLAS Monte Carlo samples [232] is an incredibly complex process and the following section takes several liberties to cover only the concepts required to understand how the neural networks fit into the flow of information, presented in Section 9.1. In brief, it can be separated into 4 steps.

1. **Generation**: A generator is first supplied with a parton distribution function (PDF), which describes the substructure of the proton. It can then simulate the hard-scatter at the parton level in an idealised theoretical environment. The output is passed through a parton shower (PS) program which also adds the underlying event (UE). Fragmentation takes place, followed by colour reconnection and hadronisation. All of these objects and steps are considered generator-level or truth-level.

2. **Detector Simulation**: The next step is to propagate all outward moving particles through a simulation of the ATLAS detector. For all datasets used in this analysis, the ATLAS detector simulation was performed using GEANT4 [232]; a high-quality simulation which incorporates detector geometry, materials, tracking through matter and magnetic fields, and the production of particle showers. The energy deposits of particles interacting with sensitive detector elements are recorded as hits.

3. **Digitisation**: In this step, the hits produced in the during the detector simulation are converted into detector signals. Digitisation represents how the experimental setup would record the traversing particles. It is here that the expected signals from pileup interactions are added to the event.

4. **Event Reconstruction**: The reconstruction procedure for simulated events is identical to the one applied to real detector signals, following the methods detailed in Chapter 6. It converts the basic signal readouts to fully calibrated objects and variables. These objects are reco-level. The collection and kinematics of reco-level and truth-level objects are not identical due to mismeasurement, particle misidentification, incomplete detector coverage, and pileup induced inaccuracies.

8.2.1 Standard Model Samples

All simulated SM processes are listed in Table 8.1, along with the name of the generator used to create them, the classes they contributed to in this project, the cross-section order, the library used for the PDFs, and various other parameter tunes and specifications. Processes generated using POWEG [233] achieved a final state

by interfacing the parton-level matrix element (ME) output to PYTHIA8 [234], which created the PS and the underlying event (UE). All samples, except those generated using SHERPA [235, 236], exploited EvtGen v1.2.0 [237] to model the decays of b- and c-hadrons. Samples involving the production of a single Z boson had a global K-factor to normalise the events to the next-to-next-to-leading-order (NNLO) QCD cross-sections.

Sample	**1)** $Z \to ee$	**2)** $Z \to \mu\mu$	**3)** $Z \to \tau\tau$
Generator	POWHEG+PYTHIA	POWHEG+PYTHIA	POWHEG+PYTHIA
Use class	Evaluation	Evaluation	Evaluation
Cross-section order	NNLO	NNLO	NNLO
PDF set	CTEQ6L1	CTEQ6L1	CTEQ6L1
PS and Hadronisation	Pythia8	Pythia8	Pythia8
Shower Tune	AZNLO	AZNLO	AZNLO

Sample	**4)** $WW \to l\nu l\nu$	**5)** $WZ \to l\nu ll$	**6)** $ZZ \to llll$
Generator	POWHEG+PYTHIA	POWHEG+PYTHIA	POWHEG+PYTHIA
Use class	Evaluation and Learning	Evaluation and Learning	Evaluation and Learning
Cross-section order	NNLO	NNLO	NNLO
ME PDF set	CT10	CT10	CT10
PS and Hadronisation	Pythia8	Pythia8	Pythia8
Shower Tune	AZNLO	AZNLO	AZNLO

Sample	**7)** $t\bar{t}$	**8)** Wt	**9)** $W\bar{t}$
Generator	POWHEG+PYTHIA	POWHEG+PYTHIA	POWHEG+PYTHIA
Use class	Evaluation and Learning	Evaluation and Learning	Evaluation and Learning
Cross-section order	NNLO	NNLO	NNLO
ME PDF set	NNPDF2.3LO	NNPDF2.3LO	NNPDF2.3LO
PS and Hadronisation	PYTHIA8	PYTHIA8	PYTHIA8
Shower Tune	A14	A14	A14

Sample	**10) (VBF)** $H \to WW$	**11)** $VV \to llll$
Generator	POWHEG+PYTHIA	SHERPA
Use class	Evaluation	Learning
Cross-section order	NNLO	NNLO
ME PDF set	CT10	NNPDF3.0NNLO
PS and Hadronisation	Pythia8	SHERPA2.2.2
Shower Tune	AZNLO	Default

Sample	**12)** $Z \to \mu\mu$	**13)** $Z \to \mu\mu$	**14)** $Z \to ee$
Generator	SHERPA	MADGRAPH+PYTHIA	SHERPA
Use class	Evaluation (alternative Z)	Evaluation (alternative Z)	Evaluation (alternative Z)
Cross-section order	NNLO	NNLO	NNLO
ME PDF set	NNPDF3.0NNLO	NNPDF2.3LO	NNPDF3.0NNLO
PS and Hadronisation	SHERPA2.2.1	PYTHIA8	SHERPA2.2.1
Shower Tune	Default	A14	Default

Table 8.1: Generators, cross-section normalisations, PDF sets and other MC tunes used in this analysis for SM processes.

Samples 1-3 in Table 8.1 model Z bosons decaying into two oppositely charged leptons of a particular flavour. These were used exclusively in the Evaluation class. The processes were generated using POWEG with a matrix element calculation at next-to-leading order (NLO) in perturbative QCD. The AZNLO [238] set of tuned UE and PS parameters were used and PDFs were taken from the CTEQ6L1 set [239].

The diboson processes which make up Samples 4-6 in Table 8.1 were generated using POWEG employing the CT10 PDF set interfacing with PYTHIA8. Events in these processes were split between the Evaluation and Learning classes.

Samples 7-9 include $t\bar{t}$ and associated top quark (Wt) production. They were generated with a POWEG NLO kernel interfaced to PYTHIA8 with the A14 set [240] of tuned PS parameters. Parton luminosities were provided by the NNPDF2.3LO PDF set [241] and for the $t\bar{t}$ sample the resummation of soft-gluon terms in the next-to-next-to-leading-logarithmic approximation with TOP++ [242] was included. The top quark mass was set to 172.5 GeV for all MC samples involving top quark production. While the associated top quark samples are inclusive, the $t\bar{t}$ sample only includes non-all-hadronic decays. These samples were also split between the Evaluation and Learning classes.

Sample 10, which models Higgs boson production via VBF, used PDFs derived from the CT10 NLO PDF set [243]. The Higgs boson mass was set to 125 GeV. This sample was used only in the Evaluation class.

Sample 11 contains additional simulated diboson samples generated with SHERPA normalised to the NNLO cross-section. SHERPA also controlled the parton showering and hadronisation. This sample employed NNPDF3.0LO PDF set [244] and was used exclusively in the Learning class. The reason for this distinction is explained in Section 9.3. To avoid confusion between this dataset and the other diboson samples, further mentions of it in this document include the '(SHERPA)' label.

Simulations of Z bosons decaying into either a pair of opposite sign muons or electrons using alternative generators are included in Samples 12-14. They were created using either SHERPA with the NNPDF3.0NNLO PDF set [244], or MADGRAPH [245] using the NNPDF2.30LO PDF set and the A14 parameter tune. Like the other Z boson samples, these were used exclusively in the Evaluation class.

8.2.2 SUSY samples

The SUSY signal processes of direct slepton production, used in Section 11, were generated from LO matrix elements with up to two extra partons. They were created using the MADGRAPH generator interfaced to PYTHIA8 with the A14 tune. The ME PDFs were provided by the NNPDF2.3LO PDF library. Jet-parton matching was realised following the CKKW-L prescription [246]. The cross-sections were calculated at NLO with soft gluon emission effects added at next-to-leading-logarithmic accuracy. The nominal cross-sections and their uncertainties were taken from an envelope of cross-section predictions using different PDF sets and renormalisation scales.

8.2.3 Pileup modelling

The ATLAS detector captures signals from additional pileup interactions which obscure the hard-scatter as described in Section 5.6. Both in-time and out-of-time pileup was modelled in all of the aforementioned MC samples. Pileup was added to the event during the digitisation step of the production.

The number of events to overlay per bunch-crossing can be set at run time and is a function of the desired luminosity to be simulated. Here it was generated from a Poisson distribution around μ, which was measured from data. Minimum bias (MB) collisions were then superimposed on top of the hard-scatter. These additional interactions were generated as low-p_T inelastic pp collisions using the soft QCD process of PYTHIA8 with the A3 tune [247] and the NNPDF2.3LO PDF set. The MC samples were reweighted so that the distribution of the number of pileup interactions matched the distribution in data.

8.3 Preselection

This section describes the preselection that was applied to all MC and data samples. This is a significant section, since the data used to train and test the neural networks all shared the following preselections. It has been observed that a trained neural network's performance becomes unpredictable when it is forced to extrapolate and attempt to fit unfamiliar data. Therefore, the results of this dissertation need to be analysed within the context of these preselections. Like the object selections discussed in Chapter 6.2, the application of the networks presented in this dissertation on new data with different preselections may be unpredictable without further training.

All datasets used in this analysis came from the JETM3 derivation used by the ATLAS Jet/$E_{\mathrm{T}}^{\mathrm{miss}}$ combined performance group. The JETM3 derivation involves a skimming filter that requires at least two baseline leptons above 20 GeV before OR.

The following requirements were taken from prescriptions provided by the ATLAS Data Preparation group and correspond to general event cleaning at ATLAS. All events are required to have at least one PV with at least two associated tracks with $p_T > 400$ MeV. Furthermore, a set of requirements was applied to both data and MC to reject events recorded during noise bursts in the ECal. Events were also removed if they were likely to contain jets from non-collision background processes, or muons created from the radiation of the ATLAS cavern or cosmic rays.

Chapter 9

Neural Network Training Process for $E_{\mathrm{T}}^{\mathrm{miss}}$

As introduced in Chapter 1, the core philosophy and goal of this project was to take the various existing $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction methods currently used by ATLAS, each detailed with their own advantages and disadvantages in Chapter 7, and use machine learning to combine them to form a single most accurate definition of $E_{\mathrm{T}}^{\mathrm{miss}}$.

The specific direction taken to achieve this goal was to train a deep and dense feed forward neural network to produce an estimate of $E_{\mathrm{T}}^{\mathrm{miss}}$ per event. Both $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitude and direction are useful observables, so the network would have to output two separate components, making this an unbounded 2D regression problem. The ideal decomposition of this vector into two components is discussed in Section 9.4.3 and 9.4.4. The model would have to be versatile and adaptive, capable of contextualising information for many different event topologies. Deep neural networks have been shown in the past to be very well suited to sophisticated multivariate regression tasks [75], and were therefore chosen over other machine learning models such as support vector machines or BDTs.

However, there exist several notable drawbacks to deep learning. Models require massive training sets, long training times, and large amounts of memory. Furthermore, in Chapter 3.7.7 it is mentioned that there is no solid theory on determining the optimal network structure. Therefore, much time was devoted towards creating and comparing different network architectures. This was a lengthy and iterative development process where almost 3000 different models were produced. To cope with such a task, significant effort was spent to increase the processing speed and to reduce the hardware impact of training. This chapter attempts to document and summarise only the most significant steps taken to develop the final working model and is structured as follows.

Section 9.1 describes the method and design principle for how the network was trained. Section 9.2 lists the specialised hardware and software used to develop the project framework. How the datasets mentioned in Chapter 6 contributed to this process is covered in Section 9.3. Discussions on how each event was observed, how

invariances were handled, and how the output was constructed, are presented in 9.4. Network training and the searches for the optimal hyperparameters are discussed in Section 3.7.7.

9.1 Training Method and Principle

The networks were trained using the standard supervised machine learning techniques covered in Chapter 3. Supervised learning was described as highly analogous to function approximation and curve fitting. An ANN attempts to approximate some multivariate mapping $f(\mathbf{x}) = \mathbf{y}$ by constructing its own function $\hat{f}(\mathbf{x}) = \hat{\mathbf{y}}$ which minimises some cost $C(\boldsymbol{\theta})$ evaluated over the training set. On a single example the loss quantifies the distance between the network's current prediction and the desired target, $L(\hat{\mathbf{y}}_i, \mathbf{y}_i)$.

This approach of using supervised learning meant that all training samples had to contain desired outputs. Hence, the networks were trained exclusively on MC simulated collisions. The target vectors were taken from the middle of the MC production chain, detailed in Section 8.2. Before the detector simulation, the total transverse momentum of all non-interacting particles was saved per event. Since this is truth-level information, it is equivalent to setting $\mathbf{y}$ as the True $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$. Select variables from the output of the event reconstruction, including the $E_{\mathrm{T}}^{\mathrm{miss}}$ working points, were used as the input vector $\mathbf{x}$. A visual representation of the function approximation is shown in Figure 9.1.

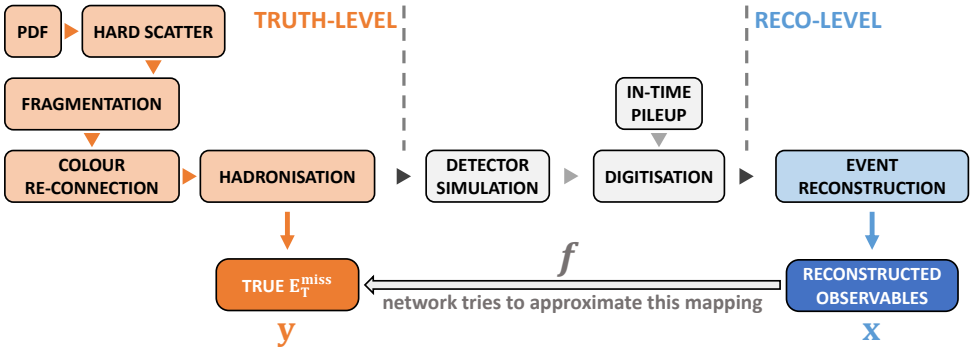


Figure 9.1: A diagram showing the simplified information flow involved in this project, starting with the basic steps of MC generation, detector simulation and finally event reconstruction. The network was only trained on MC simulated samples since it was attempting to approximate a mapping between reco-level and truth-level information.

Since the inputs were reco-level observables and are available in data, a trained network performing the function $\hat{f}$ could be applied to real collisions. This would hopefully ascertain a more accurate estimate of True $E_{\mathrm{T}}^{\mathrm{miss}}$ than the other working points. This step is represented by Figure 9.2.

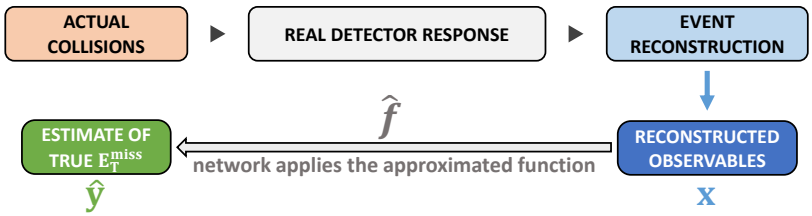


Figure 9.2: A diagram showing how a trained network could be applied to new information or real data to produce an estimate of the True $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$.

One can interpret that the function f is essentially a reversal of the reconstruction, digitisation and detector simulation steps, as shown by Figure 9.1. The reconstruction step in MC is by design identical to data, but the digitisation, detector simulation and the superposition of pileup signals in MC are themselves only approximations of reality. So a notable caveat of the approach used in this project is that it relies heavily on the accuracy of these processes. If this simulation does not reflect real life interactions, then the network's performance will be inconsistent when applied to MC and data.

While this is a point to be aware of, and indeed some of the studies later on in the next chapter do compare the relative performance of the network on real and MC signals, the GEANT4 based MC production chain used in this project is the standard at the ATLAS collaboration for accuracy. Any performance differences are expected to be small.

9.2 Hardware and Software

In this project all neural networks were developed using the PyTorch [132] deep learning library, an open-source python-based scientific computing package. PyTorch was chosen over Keras and TensorFlow due to the control given to the user and the ease that scripts could be written to take advantage of GPU hardware acceleration. The networks were either trained on a personal computer equipped with a Nvidia RTX-2070-Super graphics card or on the University of Cape Town High Performance Computing facilities, which uses Nvidia Tesla-K40m cards. The performance boosts achieved by the GPUs were invaluable. Completion of a single epoch using the largest training set took around 45 seconds, as opposed to around 15 minutes when executed solely on a CPU. This speed-up is one of the reasons that made the many comparisons between the thousands of network configurations discussed in this chapter possible.

9.3 Datasets

As mentioned in Chapter 8, a split was applied which partitioned all simulated SM datasets into a Learning class and an Evaluation class. The former was observed by

the networks during their training phase, and the latter was used both for the final evaluation of the model and as SM backgrounds in the SUSY search. This separation was necessary because neural networks are extremely prone to overfitting.

All SM datasets used in this work are listed in Table 8.1, and most datasets were used in both classes. Partitioning took place by first ordering the dataset by their MC event-number. Every third event was allocated to the Evaluation class and the remaining events were assigned to the Learning class. This ensured that the two resulting groups had similar distributions. It also meant that most datasets contributed to the Learning and the Evaluation class with a ratio of 2:1. The composition of the Learning Class is shown in Figure 9.3 and in Table 9.1.

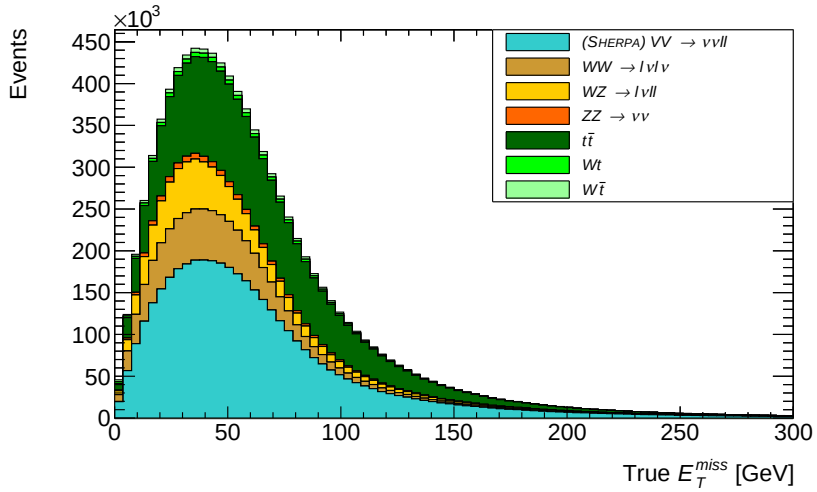


Figure 9.3: A histogram showing the composition of the Learning class where True $E_{\mathrm{T}}^{\mathrm{miss}}$ is drawn along the x-axis.

Process	**Events**	**Percentage**
(SHERPA) $VV \to llvv$	4 096 881	43.87%
$WW \to lvlv$	1 117 272	11.97%
$WZ \to lvll$	926 475	9.92%
$ZZ \to llvv$	141 833	1.52%
$t\bar{t}$	2 815 091	30.15%
Wt	119 764	1.28%
$W\bar{t}$	120 209	1.29%
Total	9 337 525	**100%**

Table 9.1: The various SM processes found in the Learning class, how many events they contributed, and that total as a percentage of the class.

It was observed that a model's performance always improved with an increase in training set size and diversity, so all available samples were used to ensure the best possible results. It is also important to note that no MC weights, reconstruction scale-factors, or pileup weights were applied to the samples in the Learning class. Each event contributed equally. This is because the realistic physical distribution of events is not a requirement for training, and it could lead to an unbalanced training

set. Many of the processes, such as $t\bar{t}$, were only represented in a single dataset. The omission of $t\bar{t}$ from the Learning class would result in a network which would perform poorly on the type of events which usually contribute the most background in QCD sensitive physics searches. Alternatively, omission from the Evaluation class would mean that the network's performance could not be evaluated in environments with high jet multiplicities, not to mention that $t\bar{t}$ events needed to be used as background in Chapter 11. Following this rationale, the $t\bar{t}$ dataset and many of the others, had to be split.

The $2:1$ ratio was chosen to favour the size of the training set. In the early stages of this project the filter was configured to produce an equal number of events in the Learning and the Evaluation class. When it was changed to its current state, a significant performance increase was observed while the Evaluation class still retained decent statistics. The imbalance between the different contributions listed in Table 9.1 are because the parent datasets themselves contained vastly different amounts of simulated events.

The diboson sample generated using SHERPA was used entirely in the Learning class since the process was already accounted for in the Evaluation class by other datasets. It also resulted in a training set which, while imbalanced with respect to the different physical processes, contained a large amount of events with typically low jet multiplicity (ZZ, ZW, WW) and a large amount of events with typically high jet multiplicity ($t\bar{t}$, Wt, $W\bar{t}$). This mixture was beneficial since the amount of jet activity in an event is one of the main features affecting $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution and response.

Three SM processes were excluded from the Learning class. The omission of $Z \to ll$ was because it was observed that the inclusion of events with True $E_{\mathrm{T}}^{\mathrm{miss}} = 0$ into the training set resulted in a network with worse performance. A network trained on such events overzealously predicted $E_{\mathrm{T}}^{\mathrm{miss}} = 0$. This empirical observation has a few potential explanations and is discussed in more detail in Section 10.3.1.

The exclusion of the VBF Higgs process was due in part to its size. The dataset was significantly smaller than all the others, offering only a few thousand events, meaning that it would contribute very little to training. Furthermore, it was used to study the network's performance on a SM process that was new to the network. Lastly, the $Z \to \tau\tau$ only became available to this project after all networks had already been trained.

9.3.1 Learning Class Event Selection

Deep learning models, particularly those used for regression, become unreliable when tasked with extrapolation. Models contain millions of parameters and do not output confidence intervals or uncertainties with their predictions. Therefore, there is no way for such regression models to indicate that a new set of inputs is unfamiliar and its corresponding outputs less certain. The tools developed in this project were

intended for use in future analyses across the entire ATLAS collaboration. To ensure the most inclusive training set, no additional event selections were used for the Learning class beyond those associated with the JETM3 derivation and the standard event cleaning listed in Section 8.3.

9.4 Network I/O

9.4.1 Input Features

One of the main decisions that had to be made for this project was what information would the network be allowed to see. The decision was influenced by several factors.

The first of these factors was the choice of network architecture; a dense feed forward neural network. This implied that the input vector $\mathbf{x}$ needed to have fixed dimensions, matching the number of input nodes. A consequence of this was that only global event variables could be used as inputs, not the individual kinematics of specific objects. If individual object features were used, then the number of inputs could differ from event to event depending on their multiplicity. This can be dealt with by overcompensating the amount of input neurons and leaving some empty if the multiplicities are low, but this is computationally wasteful. Alternatively, RNNs are sometimes used when input multiplicities vary between examples [248]. But for this project, a simpler architecture was used so all inputs had to be global variables which are well-defined for every event in ATLAS.

The network was meant to be a combination of the existing $E_{\mathrm{T}}^{\mathrm{miss}}$ definitions, so it was not in the scope of this project to redefine ATLAS object reconstruction from the ground up. This meant that no basic detector signals, such as raw calorimeter deposits or tracking information, were given as inputs. Only final and calibrated observables were used. A benefit of this approach was that uncertainties in the inputs could be propagated through the model. However, systematic uncertainties associated with the neural networks were not investigated in this work due to the limitations covered in Section 12.1.

Since the network would have to produce both magnitude and direction estimations, it needed to receive directional information as well. This meant that if an input was a vector, then its components should also be provided. The choice of basis to define these components is discussed in Section 9.4.3. Finally, the inputs had to be relevant, directly or indirectly, to $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. Some of the input features were those directly associated with the five $E_{\mathrm{T}}^{\mathrm{miss}}$ working points discussed previously, but further information was also provided to contextualise and describe the event, so that the network might make an informed decision on how to combine them.

Three categories were created, and any observable that fell into at least one of them, while still meeting the requirements stated above, was included as a network input.

1. **Observables produced by a defined $\boldsymbol{E}_{\mathbf{T}}^{\mathbf{miss}}$ working point.** The five working points used were introduced in Chapter 7. They are labelled Loose, Tight, FJVT, Calo and Track. The observables included the $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitude and ΣE_{T}. Also included to provide directional information was the vector decomposition of $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ for each working point along the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ axis, as discussed in Section 9.4.3.

2. **Unique hard- and soft-terms that make up each object-based working point.** These included the hard-terms associated with photons, electrons, muons and each type of jet collection. They also included the TST and CST. For each of these terms, four different variables were provided as inputs. From Equation 7.1, each collection contributed a vector sum of all transverse momenta. From Equation 7.3, each term contributed a scalar sum of all transverse momenta. All vectors contributed their magnitude and components using the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ vector as a basis. Since some object collections are consistent across multiple working points, only unique terms were used.

3. **Event variables which might indicate the accuracy of the $E_{\mathbf{T}}^{\mathbf{miss}}$ estimations.** These are variables which describe event topology and can provide further context to the network when it attempts to combine the different $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms. It includes those that have been observed to correlate to $E_{\mathrm{T}}^{\mathrm{miss}}$ response or resolution for at least one working point, as shown in Section 7.2.

The collection of all 65 inputs used for this project is shown in Table 9.2, 9.3, and 9.4.

It is worth noting that these inputs are, by design, not independent. Beyond correlated inputs such as N_{PV} and μ, some of the inputs can directly be calculated by combining others, such as the $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitudes and their components. In the case of event-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance all three variables found in Equation 7.4 are used as separate features. This is because, contrary to some beliefs, it is not always beneficial to ensure that the inputs are independent.

It is entirely possible that during the training process the first couple of layers of the network might recreate these combinations itself. But this takes computation space and time away from the network's primary goal of approximating the function f. It seems that if there exists some non-trivial combination of inputs that is known to be useful, like a vector sum, then it is beneficial to perform that combination manually

Input Category 1									
Tight	$E_{\mathrm{T}}^{\mathrm{miss}}$	Loose	$E_{\mathrm{T}}^{\mathrm{miss}}$	FJVT	$E_{\mathrm{T}}^{\mathrm{miss}}$	Calo	$E_{\mathrm{T}}^{\mathrm{miss}}$	Track	$E_{\mathrm{T}}^{\mathrm{miss}}$
	ΣE_{T}		$E_{\parallel}^{\mathrm{miss}}$		$E_{\parallel}^{\mathrm{miss}}$		$E_{\parallel}^{\mathrm{miss}}$		$E_{\parallel}^{\mathrm{miss}}$
			$E_{\perp}^{\mathrm{miss}}$		$E_{\perp}^{\mathrm{miss}}$		$E_{\perp}^{\mathrm{miss}}$		$E_{\perp}^{\mathrm{miss}}$
			ΣE_{T}		ΣE_{T}		ΣE_{T}		ΣE_{T}

Table 9.2: A table listing all Category 1 inputs to the neural network. The parallel and perpendicular components are with respect to the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ axis for reasons described in Section 9.4.3.

Input Category 2			
Muons	$E_{\mathrm{T}}^{\mathrm{miss},\mu}$ $E_{\parallel}^{\mathrm{miss},\mu}$ $E_{\perp}^{\mathrm{miss},\mu}$ $\Sigma E_{\mathrm{T}}^{\mu}$	Loose Jets	$E_{\mathrm{T}}^{\mathrm{miss,jet}}$ $E_{\parallel}^{\mathrm{miss,jet}}$ $E_{\perp}^{\mathrm{miss,jet}}$ $\Sigma E_{\mathrm{T}}^{\mathrm{jet}}$
Electrons	$E_{\mathrm{T}}^{\mathrm{miss},e}$ $E_{\parallel}^{\mathrm{miss},e}$ $E_{\perp}^{\mathrm{miss},e}$ $\Sigma E_{\mathrm{T}}^{e}$	Tight Jets	$E_{\mathrm{T}}^{\mathrm{miss,jet}}$ $E_{\parallel}^{\mathrm{miss,jet}}$ $E_{\perp}^{\mathrm{miss,jet}}$ $\Sigma E_{\mathrm{T}}^{\mathrm{jet}}$
Photons	$E_{\mathrm{T}}^{\mathrm{miss},\gamma}$ $E_{\parallel}^{\mathrm{miss},\gamma}$ $E_{\perp}^{\mathrm{miss},\gamma}$ $\Sigma E_{\mathrm{T}}^{\gamma}$	FJVT Jets	$E_{\mathrm{T}}^{\mathrm{miss,jet}}$ $E_{\parallel}^{\mathrm{miss,jet}}$ $E_{\perp}^{\mathrm{miss,jet}}$ $\Sigma E_{\mathrm{T}}^{\mathrm{jet}}$
TST	$E_{\mathrm{T}}^{\mathrm{miss,soft}}$ $E_{\parallel}^{\mathrm{miss,soft}}$ $E_{\perp}^{\mathrm{miss,soft}}$ $\Sigma E_{\mathrm{T}}^{\mathrm{soft}}$	CST	$E_{\mathrm{T}}^{\mathrm{miss,soft}}$ $E_{\parallel}^{\mathrm{miss,soft}}$ $E_{\perp}^{\mathrm{miss,soft}}$ $\Sigma E_{\mathrm{T}}^{\mathrm{soft}}$

Table 9.3: A table listing all Category 2 inputs to the neural network. These are the unique hard (top six) and soft (bottom two) terms that are involved in object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. The parallel and perpendicular components are with respect to the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ axis for reasons described in Section 9.4.3.

and present it to the model. Having the combination available and letting the model decide how to use it seems to lead to better results and faster training than making the model create it itself.

Category 3, presented in Table 9.4, contains a more diverse set of observables than the other two and therefore each deserves a brief mention as to why they were thought of as necessary. As previously mentioned, the philosophy behind Category 3 was to include any observables which might indicate the relative accuracy of at least one of the $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms.

The first two variables are the $E_{\mathrm{T}}^{\mathrm{miss}}$ significance estimates described in Section 7.1.3. They are defined as the likelihood that the measured $E_{\mathrm{T}}^{\mathrm{miss}}$ is still within the scale of its resolution. It is an efficient method for determining whether the calculated $E_{\mathrm{T}}^{\mathrm{miss}}$ is likely to be from fake sources only. Providing the network with these significance estimates could make it less susceptible to the positive observation bias that plagues the other working points. Even though object-based significance $\mathcal{S}_O$ was determined to be a better discriminator between events with real and fake $E_{\mathrm{T}}^{\mathrm{miss}}$ as shown in Figure 7.1, the older event-based significance $\mathcal{S}_E$ estimate is also included since it added very little computational work and no detriment was expected from its inclusion. As covered in Section 7.2, all $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms, except perhaps for Track, degrade in performance with an increase of in-time pileup. Furthermore, this degradation takes place at different rates. It is therefore crucial that if a tool was to decide on the most accurate $E_{\mathrm{T}}^{\mathrm{miss}}$ definition, it must be aware of the amount of in-time pileup present. Three different variables are included which provide estimates of in-time pileup.

Input Category 3	
$\mathcal{S}_O$	Object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance using the Tight working point
$\mathcal{S}_E$	Event-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance using the Tight working point and ΣE_{T}
μ	Mean number of interactions per bunch-crossing
N_{PV}	Number of primary vertices with at least 2 associated tracks
N_{PV^4}	Number of primary vertices with at least 4 associated tracks
N_{trk}	Number of ID tracks associated with the hard-scatter primary vertex
H_{T}	Total transverse momentum from hard-terms
$\Sigma_{FL}p_{\mathrm{T}}$	Scalar sum of p_{T} from forward Loose jets
$\Sigma_{FF}p_{\mathrm{T}}$	Scalar sum of p_{T} from forward FJVT jets
N_{FL}	Number of forward Loose jets
N_{FF}	Number of forward FJVT jets
$JetPU$	Weighted average of the jet pileup probabilities
N_J	Number of signal jets
N_e	Number of baseline electrons
N_μ	Number of baseline muons

Table 9.4: A table listing all Category 3 inputs to the neural network.

The first is the expected number of interactions per bunch-crossing μ, which uses the measured instantaneous luminosity of the colliding beams. Also included to directly indicate the number of observed pp scatters are the number of reconstructed primary vertices with at least two tracks N_{PV}, and with at least four tracks N_{PV^4}.

These three variables diverge from each other due to a phenomenon called vertex merging [230]. As in-time pileup increases so too does the likelihood that multiple pp collisions take place very close to one another. These individual scatters are not resolved by the ID and instead lead to a single vertex. This causes an underestimation of in-time pileup by N_{PV} as it increases. This underestimation is also why $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution seems to degrade more rapidly as a function of N_{PV}, rather than of μ. Both N_{PV} and N_{PV^4} experience vertex merging at different rates. Therefore, the combination of all three variables provides a more robust and versatile estimation of in-time pileup.

In addition to pileup, $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is known to degrade with an increase in event activity, hence the inclusion of H_{T}. The resolution is particularly sensitive to the amount of jet activity, which is included in Category 2 for each of the three collections of jets. But since the only difference between the three collections is their treatment of forward jets, it was thought to be beneficial to provide the network with the amount of forward jet activity and forward jet multiplicity. This information is offered in the variables $\Sigma_{FL}p_{\mathrm{T}}$, $\Sigma_{FF}p_{\mathrm{T}}$, N_{FL} and N_{FF}.

An additional source of fake $E_{\mathrm{T}}^{\mathrm{miss}}$ comes from the inclusion of jets produced by pileup interactions that were still able to pass the JVT requirement. So a multivariate discriminator was developed [224] that can indicate the probability that a jet emerged from a pileup interaction based on its p_{T}, η, and JVT discriminant. This probability is represented by P_{PU}^{jet}. To provide the network with an indication of

the total amount of jet activity expected to have come from pileup interactions, a weighted average of these probabilities was included. This is represented by $JetPU$ and all baseline jets contribute, receiving a weighting equal to their p_{T}^2 as shown by Equation 9.1.

$$JetPU = \frac{1}{\sum(p_{\mathrm{T}}^{\mathrm{jet}})^2} \sum_{\substack{\text{baseline} \\ \text{jets}}} (p_{\mathrm{T}}^{\mathrm{jet}})^2 P_{PU}^{\mathrm{jet}} \tag{9.1}$$

Also included was N_{trk}, the total number of ID tracks associated with the PV_0. This indicates the number of separate signals contributing to both the TST and Track $E_{\mathrm{T}}^{\mathrm{miss}}$. Finally the different jet, muon and electron multiplicities were also included in Category 3 by N_J, N_μ and N_e respectively as they help describe the topology of the event.

The last three variables were almost dropped over concerns that the network would use them to perform its own event selection. For example, it would be undesirable for the network to draw conclusions about the $E_{\mathrm{T}}^{\mathrm{miss}}$ of an event purely because it contained two same-flavour leptons and no others, thus being consistent with a $Z \rightarrow ll$ process. It was hoped that the network would combine the momenta of the event, not attempt to link it to a SM process it experienced during training. These inputs were only included after they were observed to have a small but positive effect on the network's final resolution even when tested on entirely new SM processes not seen during training.

This concern that the network would perform SM classification was one of the reasons that $Z \rightarrow ll$ events were not included in the Learning class. These samples have a very distinct signature and no other SM process included in the Learning class had such a focused distribution of True $E_{\mathrm{T}}^{\mathrm{miss}}$. A network would be at risk of drawing a connection between lepton flavour multiplicity and events with no True $E_{\mathrm{T}}^{\mathrm{miss}}$.

9.4.2 Output Features

The output of the networks were its estimate of the missing transverse momentum, thus constructing a new working point which is referred to as Network $E_{\mathrm{T}}^{\mathrm{miss}}$. As mentioned in Chapter 3, the format of a network output must match that of the target vectors. Therefore, each network contained an output layer comprised of two neurons corresponding to the two components of the generator-level True $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$, as shown in Table 9.5. The two components were the parallel and perpendicular projections of the vector using the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ as a basis for reasons discussed in the following section. The type of loss function used to quantify the difference between the output and the target vectors was treated as a modifiable hyperparameter, as discussed in Section 9.5.

Outputs		Targets	
Network	$E_{\parallel}^{\text{miss}}$ $E_{\perp}^{\text{miss}}$	Truth	$E_{\parallel}^{\text{miss}}$ $E_{\perp}^{\text{miss}}$

Table 9.5: A table showing the outputs of the network and the corresponding targets used for training. The parallel and perpendicular components are with respect to Tight $\mathbf{E}_{\text{T}}^{\text{miss}}$ for reasons described in Section 9.4.3.

9.4.3 Invariances

One of the primary concerns with deep learning is how to deal with invariances or symmetries in data. This topic was discussed in Section 3.7.5 which presented three alternative methods to teach symmetries to adaptive models.

In this project, the encountered invariance is a rotation about the beam axis. Since the ATLAS detector is approximately symmetric, the performance of an $E_{\text{T}}^{\text{miss}}$ working point should be consistent even if the entire event was rotated. However, to provide directional information to the neural network, many inputs are vector components. If these components were created using the standard ATLAS coordinate system then they would change with such a rotation and thus change the input values of the network. Since the structure of this model was a dense neural network, two of the methods presented in Section 3.7.5 could have been applied to teach the network this invariance.

The first would be to let the network create its own auto-encoding layer by training it on augmented data. This could be achieved by randomly rotating events between each epoch. However, this approach cannot guarantee that the resulting Network $E_{\text{T}}^{\text{miss}}$ working point displays consistent performance across the full ϕ-range. Furthermore, some studies have shown that it is much more computationally expensive and yields worse performance than the next method [134].

The second possible method involves pre-processing all input features to extract only those which are invariant under the transformation, therefore embedding the desired symmetry in the model. This was the chosen method for this project and meant that all vector components used as inputs had to be invariant under an axial rotation.

All input vectors, listed in Table 9.2 and Table 9.3, provide their parallel and perpendicular projections to an axis constructed by the Tight $\mathbf{E}_{\text{T}}^{\text{miss}}$ vector. They do not include the x and y components using the standard ATLAS coordinate system. It is for this reason the Tight $E_{\text{T}}^{\text{miss}}$ working point does not contribute such components in Table 9.2. This representation ensures rotational symmetry of all inputs while preserving relative directional information of the vectors. The Tight $\mathbf{E}_{\text{T}}^{\text{miss}}$ axis was chosen as it currently represents the default and recommended working point at ATLAS, and was thus the most valid candidate to use as a basis. The network output and target vector were also expressed in this frame.

An alternative way of interpreting this pre-processing step was that each event was rotated so that the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ vector aligned with the x-axis. This forced perspective ensured uniform performance with ϕ^{miss}. It does however mean when testing the final model, each new event had to be similarly rotated by Tight ϕ^{miss} before being given to the network. A post-processing step was also required to un-rotate the output back to the standard ATLAS frame. While the choice to do these rotations resulted in faster and more stable training than the previously mentioned auto-encoder method, the impact on which axis to choose was not investigated and could be a potential avenue of further work.

9.4.4 Polar vs Cartesian

As shown by Table 9.2 and Table 9.3 to the neural network, the vectors' directions are inferred by their parallel and perpendicular projections. If one uses the analogy of event rotation presented in the previous section, then each vector was represented by its magnitude and its Cartesian coordinates post rotation. This seems wasteful since it requires three separate, but dependent values. One might think an alternative representation would be to use just their polar angle ϕ. This would also make the pre- and post-processing steps simpler.

Early prototypes of the networks did in-fact use the polar representations for all vectors. However, regression problems directly involving angles prove to be very unstable. This stems from the fact that $-\pi$ and $+\pi$ represent the same direction. Iteration using gradient descent across this discontinuity during training is incredibly difficult. Even in single regression problems targeting an angle θ, it is better to train the model to produce the two values $\sin(\theta)$ and $\cos(\theta)$ [74] and reconstruct the angle in a manual post-processing step. After switching to Cartesian representations for all input and output vectors, training speed and stability increased, as did the final model accuracy.

9.4.5 Feature Standardisation

Each input feature was scaled using the method of standardisation introduced in Section 3.7.3. The means and variances were taken from distributions of the individual inputs using the entire Learning class. Standardisation was chosen over min-max normalisation due to the presence of outliers in many of the distributions.

Tests were performed by training three models using the same network configuration. First tests used raw data, then using standardised inputs, and finally with both standardised inputs and outputs. The last model experienced the most stable descent while also achieving the highest accuracy.

The most substantial performance gain was achieved by standardising the output. This was because the target distributions of True $E_{\parallel}^{\mathrm{miss}}$ and True $E_{\perp}^{\mathrm{miss}}$ were very distinct. Since the Tight $\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}$ was used as the basis for these projections, and since

Tight ϕ^{miss} is a decent estimator of True ϕ^{miss}, the scale of the parallel projection is much larger and more positively biased than the perpendicular projection, as shown in Figure 9.4. As the magnitude of True $E_{\text{T}}^{\text{miss}}$ increases, the Tight working point becomes a better estimator of its direction. This diminishes the perpendicular projection and increases the positive parallel projection of the two vectors, hence the existence of the long positive tail along the x-axis. This meant that during training using gradient descent, it was numerically more favourable to optimise the $E_{\parallel}^{\text{miss}}$ node rather than the $E_{\perp}^{\text{miss}}$ node. After standardisation the target distributions are brought closer together in scale giving each more equivalent importance.

Using standardisation required that the means and variances calculated from the Learning class had to be saved alongside the final model. When applying a trained network to new data, the data would first have to be scaled using the saved values, along with the rotations discussed in Section 9.4.3. The output of the network was post-processed similarly, by first undoing the standardisation and then un-rotating the event.

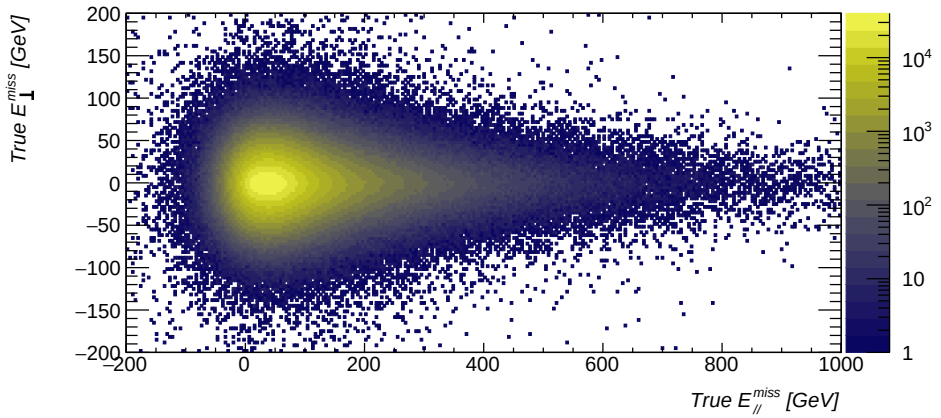


Figure 9.4: A heat-map showing the distributions of the parallel and perpendicular component of the genuine missing transverse momentum with respect to Tight $\mathbf{E}_{\text{T}}^{\text{miss}}$. Therefore, this is also a heat-map of the target vectors used to train the networks. The heat-map includes all events from the Learning class. As shown by the colour bar, a log scale is used for the z-axis.

9.5 Optimal Network Structure

As discussed in Section 3.7.7, a model's performance is greatly dependent on its hyperparameters, which describe network configuration and are set before training. Thus in every deep learning application, an effort should be made to test different combinations of hyperparameters to fully develop the most accurate model. However, due to the quantity of different hyperparameters, the number of values each can take, and therefore the massive number of possible network configurations, searching for the optimal network is an intensive and iterative process.

In this project the search was broken into two main segments. The first involved an automated grid search changing the basic network structure and descent algorithms. The observations from this first segment were used to direct a second round of tests into network structure, regularisation methods and more.

To save time, many hyperparameters were not changed and were kept at their default values. For example, while the choice of optimiser was varied and algorithms such as Nesterov momentum and Adam were compared, the internal hyperparameters of those methods were not changed. The momentum coefficient was kept at $\gamma = 0.9$ and the decay rates of Adam were left as $\gamma_1 = 0.9$ and $\gamma_2 = 0.999$. Therefore, if the hyperparameters of a method or activation function are not mentioned in this section, one can assume that they were kept at their default values listed where the methods are covered in Chapter 3.

Also kept constant throughout this investigation was the activation function, or lack thereof, in the final output layer of the networks. Since the task is an unbounded regression problem, it is convention to have simple linear units (with biases) in the output layer. Therefore, only the hidden neurons in these networks contained non-linear activation functions and to further simplify the model only one type of activation function was used across a single network. All trainable parameters were initialised using the method described in Section 3.7.4, including the special case for when the SELU activation function was in use. Finally, all hidden layers shared the same width, allowing each network size to be described by $(\text{depth} \times \text{width})$.

Ideally, each of these networks would be trained on the full training set and evaluated using cross-validation on an orthogonal testing set. However, even with hardware acceleration enabled and significant effort being spent in program optimisations, it took around 100 hours to fully train a single network using all events. The number of networks trained in just the initial grid search totalled 2880, meaning that a full study would have taken over 30 years of computation time, so some compromises had to be made.

The most significant compromise was that the training set size used in the hyperparameter searches had to be greatly reduced. The consequences of this was that all conclusions drawn from these searches would be less certain, especially concerning the best methods to combat overfitting. It also meant that some of the larger networks in the grid search contained more parameters than training examples, something that is strongly discouraged in any machine learning project. Care was taken to focus mainly on general learning trends rather than full optimisations.

9.5.1 Initial Grid Search

An automated grid search was performed and all hyperparameters with their possible values/functions are listed below. A new model was trained for almost every possible combination.

- The network width: 100, 200, 500, **1000**, 2000.
- The network depth: 3, **5**, 7, 10.
- The learning rate η: 10^{-2}, 10^{-3}, 10^{-4}.
- The loss function L: MSE, MAE, **Huber** (Section 3.5).
- The optimiser: NAG, **Adam**, AdaMax, AdaDelta (Section 3.4).
- The activation function σ: ReLU, PReLU, ELU, SELU, **Swish** (Section 3.3).

The only configurations excluded in the grid search were the networks with widths of 1000 or 2000 and a depth greater than five. This was due to the devices running out of memory. The three chosen values for the learning rates were selected after an initial round of tests to find ones which resulted in a stable descent for most configurations. The learning rates used in this search did not carry over to the next stage since the descent and batch sizes were vastly different, not to mention the inclusion of regularisors required further adjustment to η.

To increase training speed, the networks performed full batch gradient descent. Furthermore, the training set for this search was a randomly selected subset of the Learning class containing only 100 000 events. This was in order to save both time and memory. One of the consequences of using such a small training set was that overfitting was unavoidable. This is one of the reasons that the tests for the optimal regularisor was performed in the next search. Instead, no early stopping or cross-validation was employed, and the networks were judged purely on their performance on the training set. This was acceptable since none of these networks were used as a final model and this was purely a test to find the combination of size, optimiser, loss function, and activation function that led to the most non-trivial learning.

Since different loss functions were used in this search, the networks were compared to one another using the RMSE calculated over all 100 000 events. The network which produced the lowest RMSE at any point during its training process was flagged. Training was stopped after either 4000 epochs had been completed or 20 minutes had passed. Two networks were trained concurrently, one on a personal computer and the other on the UCT High Performance Computing cluster. Even with all of these reductions and optimisations the entire search took around 20 days to complete.

The training profiles of the top eight performing networks are plotted in Figure 9.5. All eight were trained by minimising Huber loss using the Adam algorithm. The

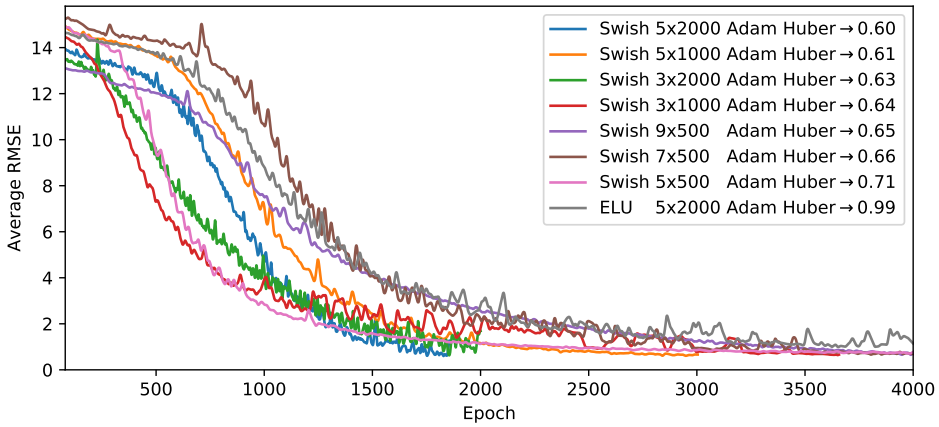


Figure 9.5: The training profiles of the top eight performing neural networks in the grid search, based on the RMSE averaged over a training set of 100 000 events. The legend shows the configurations of the networks in terms of the: activation function, size, optimiser and loss function. Also included in the legend was the minimum RMSE achieved by each model.

top seven used the same Swish activation function for all hidden neurons. Table 9.6 shows the lowest RMSE achieved by a network at each size configuration. The optimal performing network at each size used the same combination of the Swish function, Huber loss and the Adam algorithm.

That this combination led to the best performance overall was not a particularly surprising result. The Swish activation function was developed purely due to empirical evidence of its learning capabilities [112], and the Adam optimiser is one of the most widely used algorithms in machine learning today [140]. In addition, the Huber loss function excels at regression problems where the data is both noisy and contains outliers, exactly this type of task. What is surprising is how consistently this trio outperformed all other competition. Going forward, unless explicitly stated, all other networks discussed in this project were developed with these three features.

Since this test did not use cross-validation, it significantly favoured larger networks with more trainable parameters, as it almost encouraged overfitting. From Table 9.6, it is evident that an increase in network size did indeed improve performance, particularly in increasing network width*. However, the results can still indicate a potential upper limit on the required complexity or size of the models, beyond which little performance gain is achieved. Training accuracy saturated at around RMSE $\approx$ 0.60. Network performance did not increase notably when the width was increased from 1000 to 2000, despite representing a quadrupling of the number of trainable parameters, which had a major impact on computational cost and training speed. Similarly, for the networks 500 neurons wide, increasing the depth beyond five hidden layers had very little impact on accuracy. These observations however are also probably dependent on the dataset size.

*The number of trainable parameters in a fully connected network increases linearly with depth but quadratically with width.

		Network Depth			
		3	**5**	**7**	**9**
Network Width	**100**	12.49	10.13	8.55	8.03
	200	8.81	3.73	2.65	2.01
	500	1.61	0.71	0.66	0.65
	1000	0.64	0.61	X	X
	2000	0.63	0.60	X	X

Table 9.6: The minimum RMSE values reached by the best performing networks in the grid search at each size configuration. All of these networks used the Adam optimiser to minimise Huber loss and used the Swish activation function.

9.5.2 Secondary Optimisations

Following the grid search, further tests were run to check the effects of various regularisors and other deep learning techniques. Regularisors, introduced in Section 3.6, strive to prevent overfitting. These tests followed some structure, but the entire process was a form of GSD, explained in Section 3.7.7. Since this was a manual process, the learning rate could be individually adjusted for each network to ensure both stable yet fast learning.

Since many network configurations still had to be compared, the full Learning class could not be used due to time constraints. Instead, the training set was generated from a random sub-sample of one million events. All networks in this section had the aforementioned combination of Swish function, Huber loss and the Adam algorithm, as this was one of the more certain conclusions which could be provided by the initial grid search. The network width was initially set to 1000, and its depth to 5, but this was later reduced. An orthogonal testing set of 500000 events was similarly drawn to monitor overfitting using cross-validation and the holdout method, detailed in Section 3.6.1. The patience for the holdout method was set to 100 epochs, but training was also stopped if the entire process exceeded 10 hours. Networks were compared by the lowest loss achieved on the testing set.

Networks with dropout applied after each hidden layer were tested using different p values. Both L^1 and L^2 parameter norm penalties were applied separately in combination with α values equal to either 0.05 or 0.1. As explained in Section 3.7.6, BatchNorm can provide some regularisation effects while also improving training capabilities. Thus, additional networks were trained using a BatchNorm layer after each hidden one. Except for a single instance where a network applied BatchNorm on its first three hidden layers and dropout on its final two, each of these regularisors were applied separately.

The network configurations and their final testing losses are listed in Table 9.7. The default network with no built-in regularisation techniques achieved a minimum testing loss of 0.20318 after 1173 epochs, taking just over 100 minutes to train. Three other networks failed to improve on this score; the network employing BatchNorm and two networks employing large parameter norm penalties. It seems that for this

Regularisation Method	**Best Testing Loss (Huber)**
Dropout: $p = 0.2$	0.20177
Dropout: $p = 0.3$	0.20178
Dropout: $p = 0.1$	0.20194
Dropout: $p = 0.5$	0.20295
BatchNorm & Dropout: $p = 0.3$	0.20313
L^2: $\alpha = 0.05$	0.20305
L^1: $\alpha = 0.05$	0.20321
NONE	**0.20318**
BatchNorm	0.20326
L^2: $\alpha = 0.1$	0.20340
L^1: $\alpha = 0.1$	0.20345

Table 9.7: The best performing networks, each with size 5×1000, during the secondary search with their regularisation methods.

configuration, the noise and randomness induced by BatchNorm failed to produce an adequate normalisation method. However, it is worth noting that it greatly increased the rate of convergence during training, reaching its minimal loss after only 173 epochs, corresponding to around 15 minutes. Dropout was by far the superior technique and the highest achieving network used a dropping probability of $p = 0.2$, though it did increase the required training time to 2000 epochs, or just under three hours.

After this second round of results, further - albeit less structured - testing commenced. However, no notable improvement was made. The learning rates were tweaked, different schemes of parameter initialisation were employed, and non-dense architectures were used. Some networks were trained using the Euclidean distance as a loss function. Both network depth and widths were once again varied, but this only reflected the observations seen in the first round of testing; that a smaller network was noticeably worse, and a larger one offered very little improvement at significant computing cost.

Also trained was a particularly deep neural network equipped with the SELU activation function and a specialised dropout technique called alpha-dropout [111]. It is important to reiterate that these tests were done using cross-validation accuracy, so did not favour larger networks that overfit like the initial grid search. So even though the size configuration of 5×1000 led to more parameters than samples in the training set, equipped with the right regularisor, it consistently led to higher cross-validation accuracy than any network of smaller size.

The mini-batch size was found to have negligible impact on the final performance and was thus optimised for training speed. All networks were trained with a mini-batch size of 4000. During this time the tests mentioned in previous sections (such as observing the effects of input standardisation and the inclusion of lepton multiplicity) were performed.

Not every deep learning technique was tried of course, and some which may yet yield improved performance include layer normalisation [249], drop-connect [250] and layer pooling. But for a project with a limited timeframe, it was deemed that the searches conducted were satisfactory and the final network design could be considered reasonably optimised.

9.5.3 Final Model Features and Training

Based on the results discussed in the previous section, the final network was configured with five hidden layers, each with 1000 neurons equipped with the Swish activation function. It was trained using the Adam algorithm to minimise Huber loss. A dropout layer with $p = 0.2$ was applied after each hidden one.

To train the final model, the entirety of the Learning class was utilised either in the training set or the testing set for cross-validation. The testing set was created by randomly selecting 500 000 events from the Learning class. This number was chosen since it was large enough to result in a good representation of the overall class, while still leaving most of the events available to be directly trained on. The training set contained the remaining 8 837 525 events. The patience for the holdout method was set to 500. The training profile of the final model is shown in Figure 9.6 and the entire process took around 100 hours. No instability is present and saturation of the testing loss is clearly visible, indicating the moment where overfitting took over. The state at epoch 8028, which corresponded to the lowest testing loss, was saved for all future use.

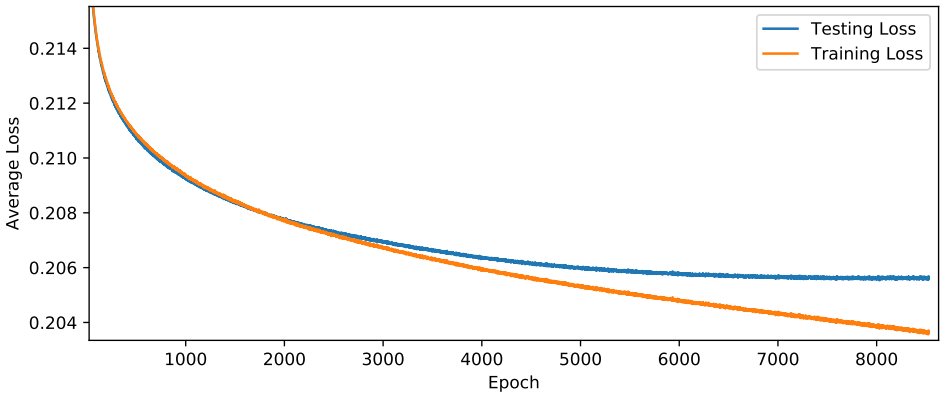


Figure 9.6: The training profile of the final network, showing the average Huber loss calculated on both the testing and the training set after each epoch.

Chapter 10

Performance of Network $E_{\mathrm{T}}^{\mathrm{miss}}$ in Data and MC Simulation

This chapter offers an overview of the performance of the newly defined Network $E_{\mathrm{T}}^{\mathrm{miss}}$ working point. It is based on the final model trained in Section 9.5.3. The standard techniques used to evaluate ATLAS's existing $E_{\mathrm{T}}^{\mathrm{miss}}$ definitions are presented in Section 7.2. That section discusses why each metric is considered and how the performance of an $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm can vary wildly when studied in different event topologies. It also explains how the techniques differ depending on the expectation values of the final state and whether or not the study used data or purely simulated samples. Metrics for $E_{\mathrm{T}}^{\mathrm{miss}}$ performance include resolution, angular resolution, tail distribution and response. All the concepts from Section 9.5.3 are again presented here in the context of this new working point.

Several signal regions were used for performance evaluation. These regions and corresponding event selections largely match those used in previous $E_{\mathrm{T}}^{\mathrm{miss}}$ performance studies [9–11]. Every MC event included in these studies was taken from the Evaluation class. Real data was used only in the $Z \to ll$ region, which also included MC contributions from the signal process and several expected backgrounds. Studies in all other regions were performed purely on MC simulation.

10.1 $E_{\mathrm{T}}^{\mathrm{miss}}$ in Final States Without Neutrinos

The $Z \to ll$ final state is particularly useful for studying the effects of fake $E_{\mathrm{T}}^{\mathrm{miss}}$. The expectation value of True $E_{\mathrm{T}}^{\mathrm{miss}}$ for this process is zero, so $E_{\mathrm{T}}^{\mathrm{miss}}$ performance can be measured in both MC and data. The region can also be further split into two channels depending on the flavour of the leptons. It is important to note that the neural network was deliberately not trained on $Z \to ll$ events, so this section also demonstrates its ability to generalise.

The inclusive signal region required the following:

- The event contained exactly two same-flavour opposite-sign (SFOS) signal leptons, with no other baseline leptons present.
- The lowest unprescaled single lepton (electron or muon) trigger was fired and at least one of the leptons matched the trigger.
- The leading lepton was required to have $p_{\mathrm{T}} > 30\,\mathrm{GeV}$ while the sub-leading needed $p_{\mathrm{T}} > 20\,\mathrm{GeV}$.
- The invariant mass of the di-lepton pair was within 15 GeV of the Z boson mass estimate: $\mathrm{m}_Z = 91.1876\,\mathrm{GeV}$.

10.1.1 Agreement between MC and Data

As discussed in Section 9.1, one of the limitations of this project stems from the fact that the neural networks were trained exclusively on MC simulations. They therefore depend heavily on the accuracy of those simulations in order to have consistent performance on both MC and data. In this section this consistency is scrutinised individually for the electron and the muon channel of the inclusive $Z \to ll$ final state, where all expected SM backgrounds were modelled.

Each MC event was weighted by its corresponding MC and pileup weights, as well as scale factors for the filter efficiency, lepton selection and trigger. Each process was normalised using their cross-sections to the integrated luminosity of the data. Some datasets were also upscaled if they were split into the Learning and Evaluation classes. To focus on differences in distribution shapes rather than statistical discrepancies, the MC events received a small correction factor so that the total number of weighted events matched the number of real collisions*, in line with the methodology presented in Reference [10].

As is explained in Section 12.1, the propagation of systematic uncertainties through a neural network was not covered in this project. Therefore, MC uncertainty stemmed from three sources only: statistical uncertainty, uncertainty on the total luminosity measurement of $43\,\mathrm{fb}^{-1}$ and the uncertainty associated with the cross-sections.

Also presented in this section is how the neural network performed on MC samples created using different generators of the same process. Alternative $Z \to \mu\mu$ samples were created using POWHEG, SHERPA and MADGRAPH. Alternative $Z \to ee$ samples were created using POWHEG and SHERPA. The name of the generator is present on each of the following graphs.

*This correction factor was around 4% for the muon channel and 3% for the electron channel.

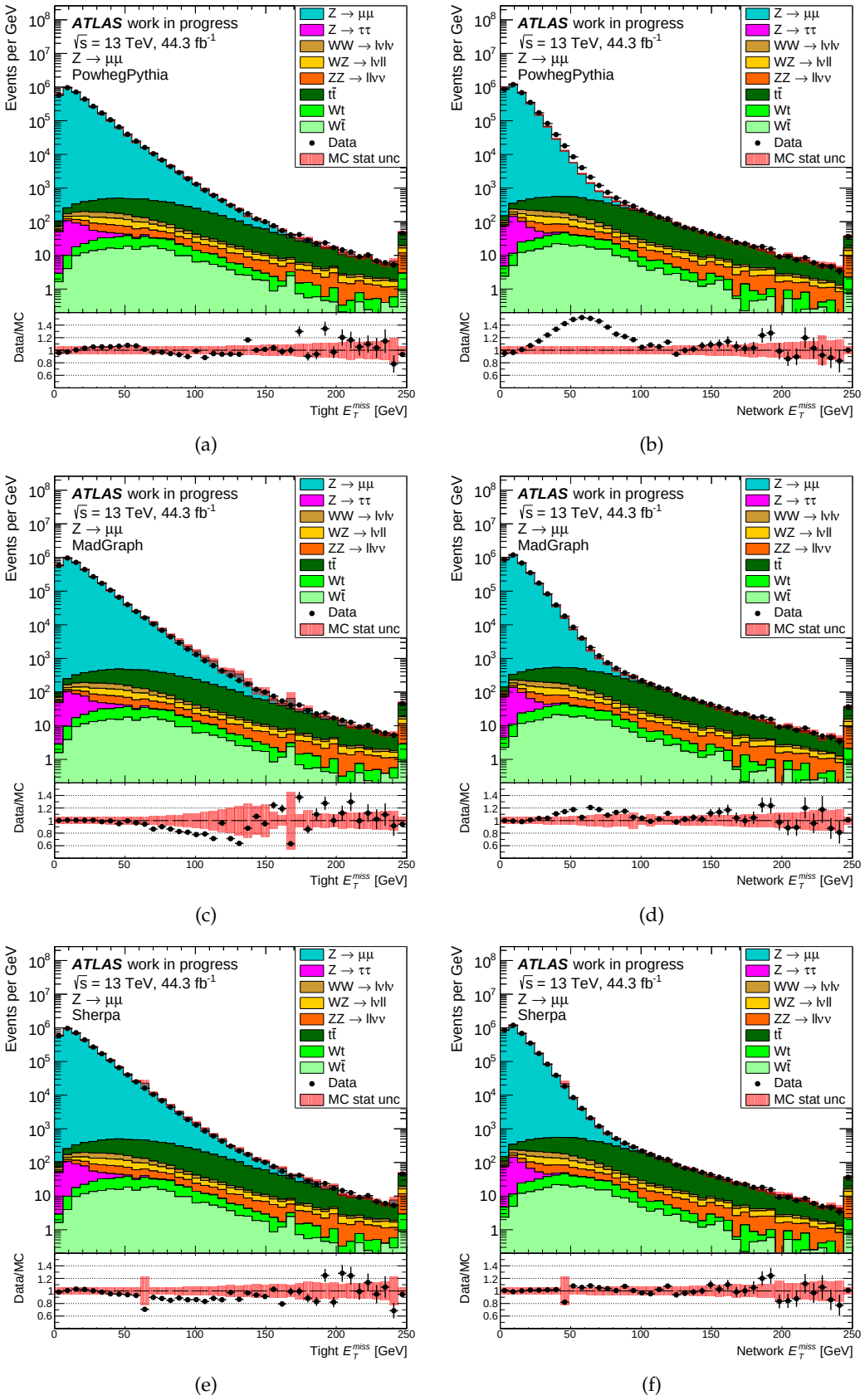


Figure 10.1: Distributions of $E_{\mathrm{T}}^{\mathrm{miss}}$ using the (a) Tight and (b) Network working points for an inclusive sample of $Z \to \mu\mu$ events, where the MC signal sample was generated using POWHEG. The same distributions using MADGRAPH are shown in (c) and (d), and again using SHERPA in (e) and (f). The last bin of each plot includes the overflow.

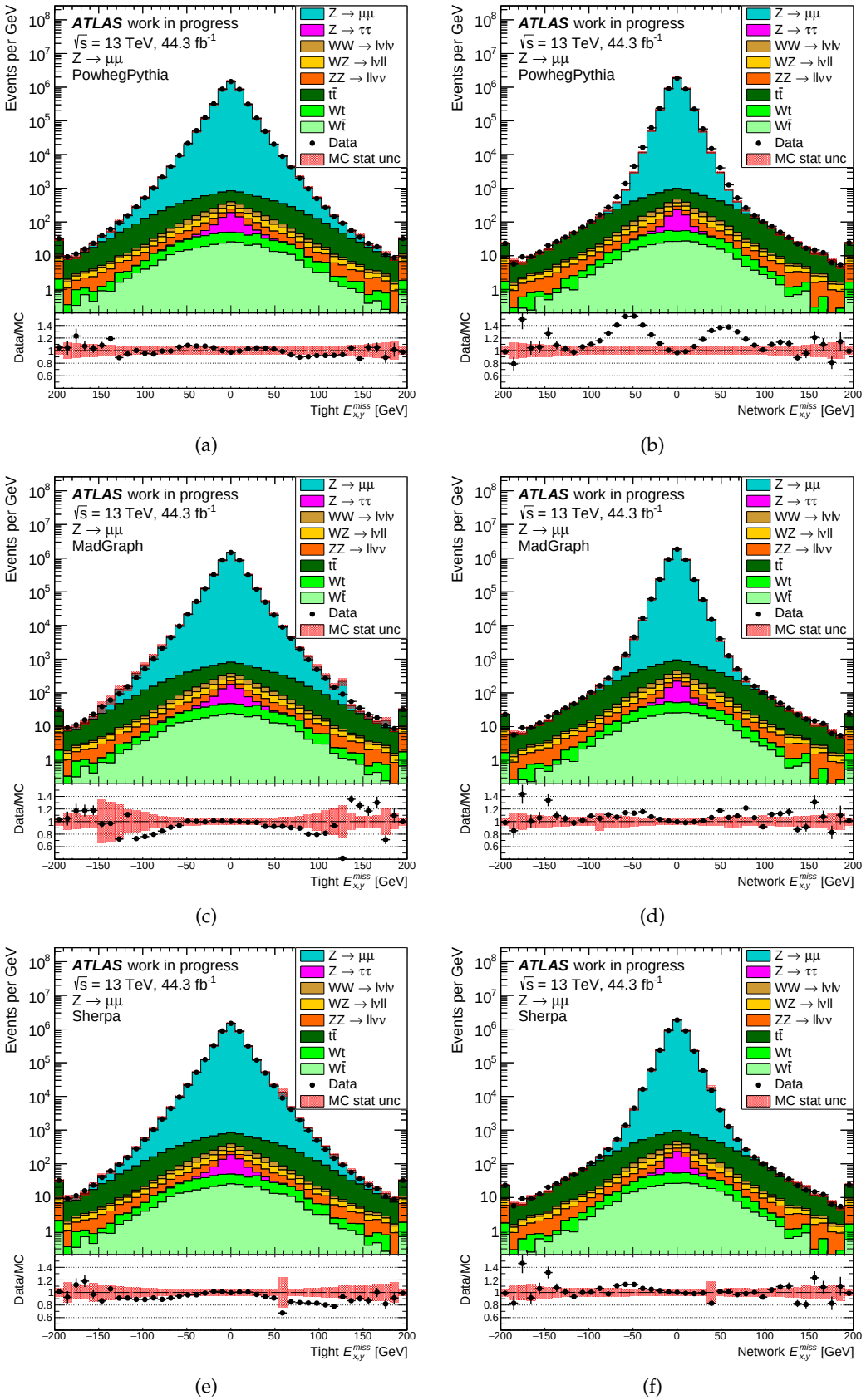


Figure 10.2: Distributions of (a) Tight $E_{x,y}^{\text{miss}}$ and (b) Network $E_{x,y}^{\text{miss}}$ for an inclusive sample of $Z \to \mu\mu$ events, where the MC signal sample was generated using POWHEG. The same distributions using MADGRAPH are shown in (c) and (d), and again using SHERPA in (e) and (f). The first and last bin of each plot includes the underflow and overflow, respectively.

The quality of the MC modelling of $E_{\mathrm{T}}^{\mathrm{miss}}$ and $E_{x,y}^{\mathrm{miss}}$ (the combined distribution of E_x^{miss} and E_y^{miss}) was evaluated by comparing the distributions of these observables to data. This was done for both the Tight and the Network working points. The results for the muon channel are presented in Figure 10.1 and Figure 10.2. The results for the electron channel were very similar and so are located in Appendix A in Figure A.1 and Figure A.2.

All distributions of Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ and $E_{x,y}^{\mathrm{miss}}$ display decent agreement between data and MC, within 20% for the bulk. Many differences are within statistical uncertainties, and those outside are expected to be covered by the JES systematic uncertainty. This is supported by graphs in Reference [10], which show the same observables, albeit with different data, as those presented in Figures 10.1(a) and 10.2(a). All corresponding graphs display equivalent ranges in the ratio plots. The distributions created using the Network working point show around the same level of agreement between data and MC for all plots except where POWHEG was used. In this instance, discrepancies in the range 25 GeV $<$ Network $E_{\mathrm{T}}^{\mathrm{miss}} <$ 100 GeV, as shown in Figures 10.1(b) and 10.2(b), are significantly larger. The reason for this is discussed below, but it is believed to be due to POWHEG mismodelling some of the variables used as inputs to the neural network.

Nearly all signal events have zero genuine missing transverse momentum. Therefore, an accurate $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm would produce a signal distribution centred on zero with very little variance. Any deviations from zero or an increase in the variance in the signal distribution can be attributed to fake $E_{\mathrm{T}}^{\mathrm{miss}}$. Compared to Tight, the Network working point produced signal distributions more tightly focused on zero while the background distributions, which do possess non-zero True $E_{\mathrm{T}}^{\mathrm{miss}}$, were left largely unchanged. This offered a more distinctive signal-vs-background separation and indicated a more accurate reconstruction. For the samples generated by POWHEG, the variance of the MC signal was decreased more than the variance of the corresponding data.

The neural network was trained to reverse the processes of MC digitisation and detector simulation, so the accuracy of these steps has a clear impact on how consistently it could be applied to data. The generator was not a part of this process, and therefore it should not affect the type of mapping learned by the neural network. However, the generator does influence the final state of the event and therefore generator mismodelling can result in differences between the data and MC distributions of the inputs to the network. A fully trained network will display disagreement between data and MC in its output distribution if there was already disagreement in any one of its input distributions.

Since notable disagreement was observed in the POWHEG samples, but not the ones created by SHERPA and MADGRAPH, a study was performed to find which of the 65 inputs was mismodelled. The most significant culprit was found to be the jet

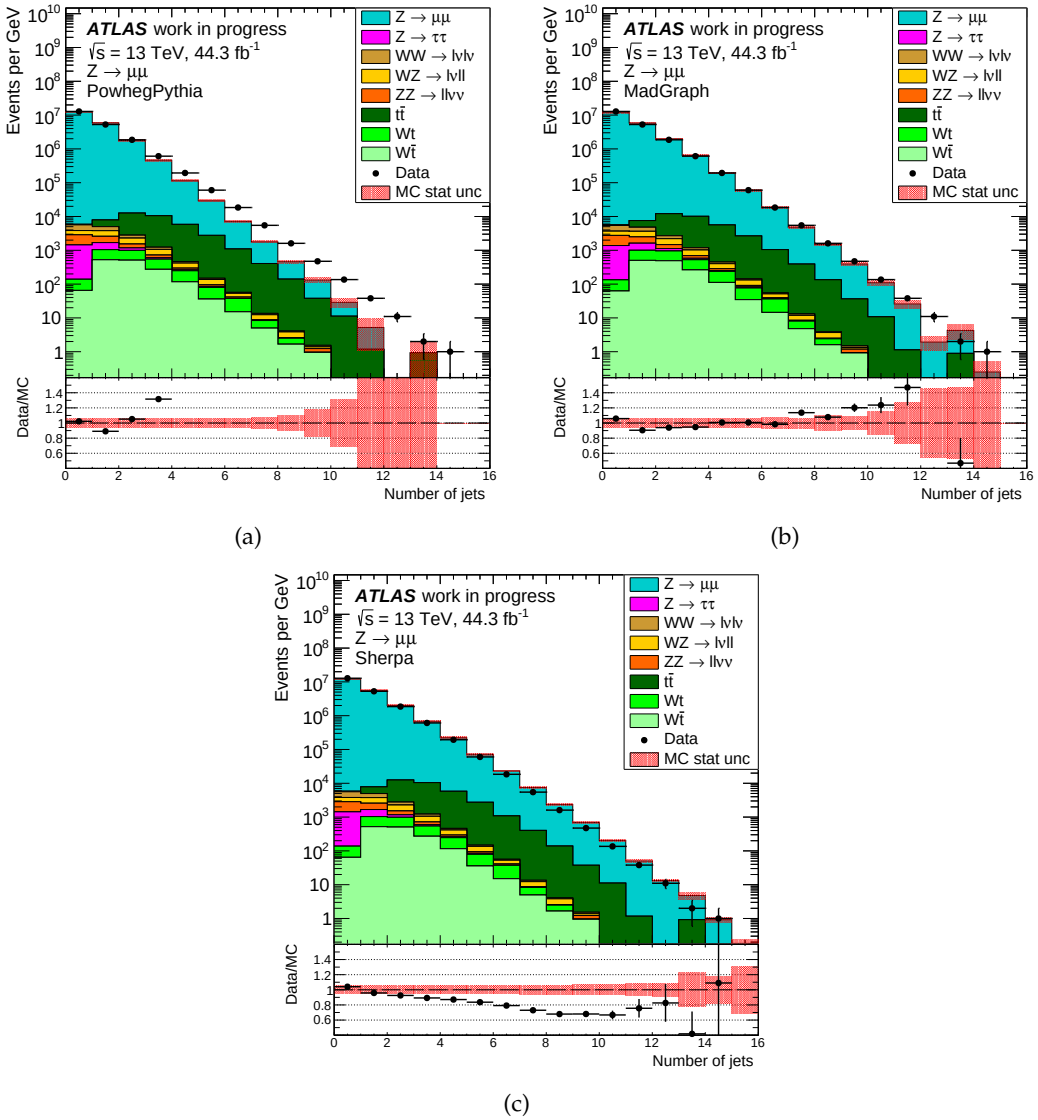
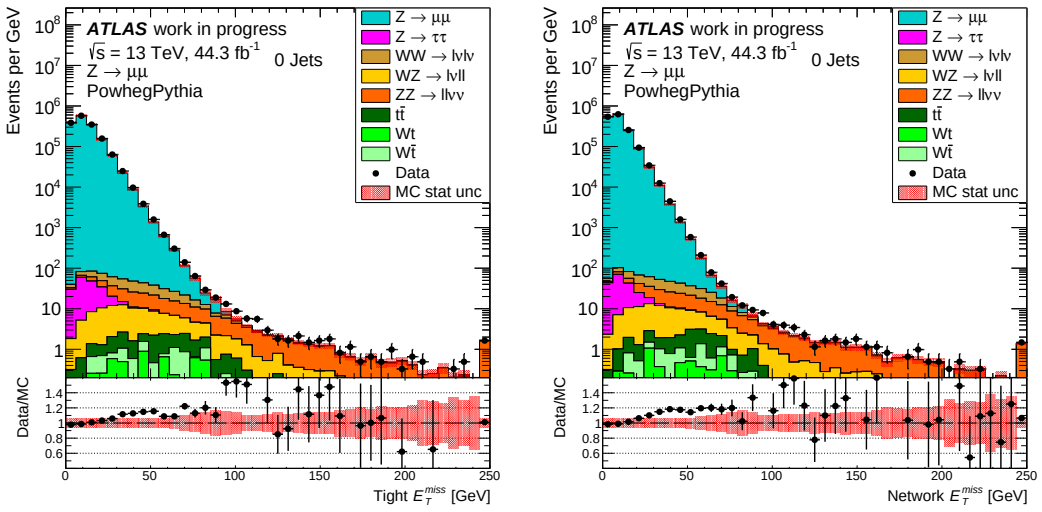


Figure 10.3: Distributions of the jet multiplicity for an inclusive sample of $Z \to \mu\mu$ events where the signal events were generated using (a) POWHEG, (b) MADGRAPH and (c) SHERPA. The shaded areas indicate the total uncertainty from the cross-sections, the measured luminosity in data, and the statistics only.

multiplicity, as shown in Figure 10.3 for the muon channel and in Figure A.3 for the electron channel. MADGRAPH and SHERPA have additional matrix element jets, while POWHEG does not and can therefore not predict high jet multiplicities in Z events accurately. The MC samples in Figures 10.1(b), 10.2(b) contained fewer jets than what was present in data. In this cleaner event topology, more MC events were reconstructed closer to the expectation value of zero. This resulted in less MC events with large fake $E_{\mathrm{T}}^{\mathrm{miss}}$ and an underestimation of MC events in the tail. On the other hand, MADGRAPH and SHERPA modelled the jet environment more accurately and did not lead to this discrepancy.

That the jet multiplicity was the primary cause of the inconsistency is further supported by Figure 10.4, which is of an exclusive $N_{\mathrm{jet}} = 0$ region using the POWHEG

sample. Here the level of agreement between data and MC is consistent between the two working points.



(a) (b)

Figure 10.4: Distributions of $E_{\mathrm{T}}^{\mathrm{miss}}$ using the (a) Tight and the (b) Network working points for an inclusive sample of $Z \to \mu\mu$ events with no jets. The shaded areas indicate the total uncertainty from the cross-sections, the measured luminosity in data, and the statistics only. The last bin of each plot includes the overflow.

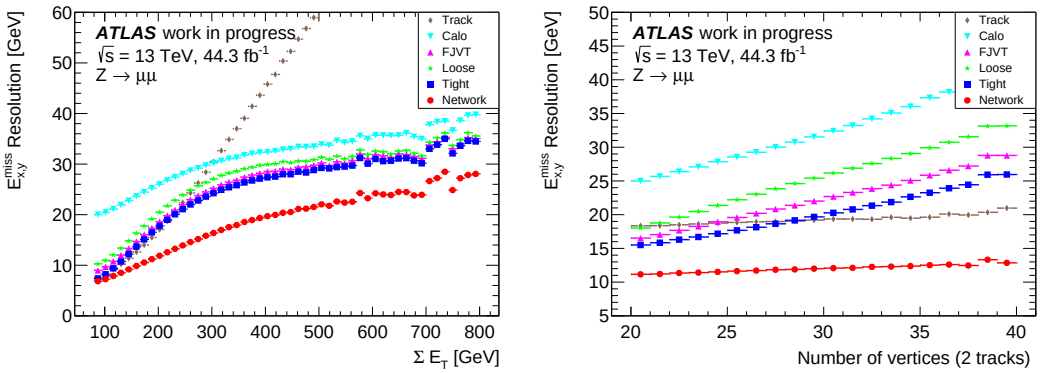
So despite the fact that the network was able to transfer well to data, any mismodelling in the input distributions can be amplified, and could lead to mismodelling in the overall Network $E_{\mathrm{T}}^{\mathrm{miss}}$ distributions. Therefore, the inclusion of variables that are difficult to model, such as N_{jet}, into the list of network features should be given extra thought. An interesting avenue of future work would be to retrain the network with the variable excluded, and see how it differs in terms of performance and modelling capabilities. This is discussed further in Section 12.1.

For the remainder of this document, all $Z \to ll$ events were modelled using SHERPA. It is important to note that the network was trained using samples generated by both POWHEG and SHERPA, as shown in Table 8.1, and thus experienced events with and without additional matrix element jets.

10.1.2 Resolution

The resolution of Network $E_{\mathrm{T}}^{\mathrm{miss}}$ was compared to the other working points in $Z \to \mu\mu$ events extracted from data. Furthermore, studies were performed on an inclusive sample and on a subset of events with no jets. Similar results were produced for the electron channel and are located in Appendix A. The resolution was measured by RMSE as defined by Equation 7.15, and evaluated as a function of the event activity. For consistency, when comparing the $E_{\mathrm{T}}^{\mathrm{miss}}$ methods, the event activity was measured using ΣE_{T} as defined by the Tight working point only.

Figure 10.5(a) shows that the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ produced the best resolution across the full ΣE_{T} range. At the minimal ΣE_{T} of around 80 GeV, the neural network showed similar performance to the other pileup suppressed algorithms. This is because most events at that scale contain no jets, the primary source of fake $E_{\mathrm{T}}^{\mathrm{miss}}$. In the range $80\,\mathrm{GeV} < \Sigma E_{\mathrm{T}} < 200\,\mathrm{GeV}$ all working points show a somewhat linear rise in RMSE. In this range, the two muons are the dominant terms contributing to $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction and possess a p_{T} resolution proportional to $(p_T^{\mu})^2$. Also, in this range Network $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution notably degrades less with an increase of ΣE_{T} compared to the other working points. For $\Sigma E_{\mathrm{T}} > 400\,\mathrm{GeV}$ the $\Sigma E_{\mathrm{T}}^{\mathrm{jet}}$ term is dominant and the difference in the resolution of Network $E_{\mathrm{T}}^{\mathrm{miss}}$ and Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ becomes constant.

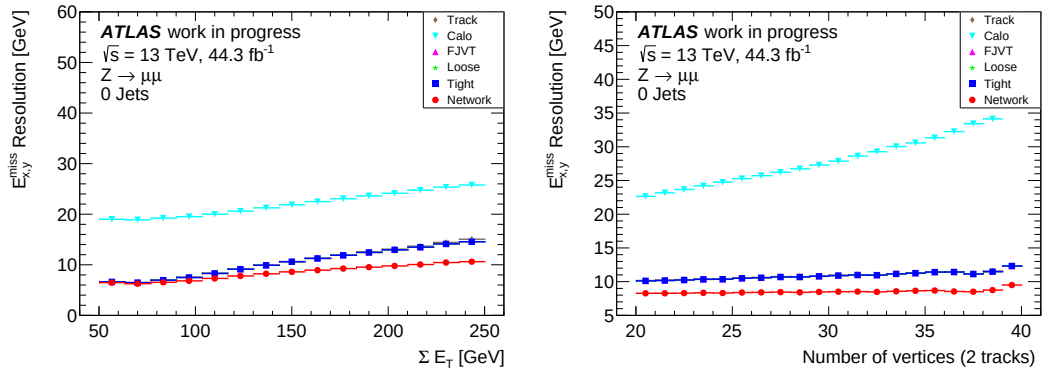


(a) (b)

Figure 10.5: The resolutions of six $E_{\mathrm{T}}^{\mathrm{miss}}$ working points, in (a) bins of Tight ΣE_{T} and (b) bins of the number of reconstructed primary vertices, in an inclusive sample of $Z \to \mu\mu$ events extracted from data.

Figure 10.6 shows the dependence of RMSE on ΣE_{T} in $Z \to \mu\mu$ events with no jets using four $E_{\mathrm{T}}^{\mathrm{miss}}$ working points. Loose and FJVT $E_{\mathrm{T}}^{\mathrm{miss}}$ are not visible since they are equivalent to Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ in the absence of jets. For this sample, the dominant source of fake $E_{\mathrm{T}}^{\mathrm{miss}}$, other than the muon p_{T} resolution, is the incomplete reconstruction of the hadronic recoil which is primarily captured by the soft-terms. Here the performance gain produced by the neural network over Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ increases approximately linearly with ΣE_{T} in the range $\Sigma E_{\mathrm{T}} > 80\,\mathrm{GeV}$. However, below that range, the performance of the neural network is nearly equivalent to Tight and Track $E_{\mathrm{T}}^{\mathrm{miss}}$. This graph and the one shown in Figure 10.5(a) indicate that Network $E_{\mathrm{T}}^{\mathrm{miss}}$ is more adept at capturing high energy hadronic recoil if it has not been reconstructed as a fully calibrated jet.

Figures 10.5(b) and 10.6(b) show the dependence of resolution on in-time pileup measured by N_{PV} in the inclusive and 0-jet samples, respectively. For the inclusive sample, Network $E_{\mathrm{T}}^{\mathrm{miss}}$ once again outperformed all other methods across the full range and demonstrated excellent stability against pileup with little to no degradation. The Network RMSE shows a gradient similar to Track, but around 15 GeV lower. In the 0-jet sample, the Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is also virtually independent of



(a) (b)

Figure 10.6: The resolutions of six $E_{\mathrm{T}}^{\mathrm{miss}}$ working points in (a) bins of Tight ΣE_{T} and (b) bins of the number of reconstructed primary vertices on a 0-jet sample of $Z \rightarrow \mu\mu$ events extracted from data. The Tight, FJVT and Loose $E_{\mathrm{T}}^{\mathrm{miss}}$ methods have identical performance in this final state.

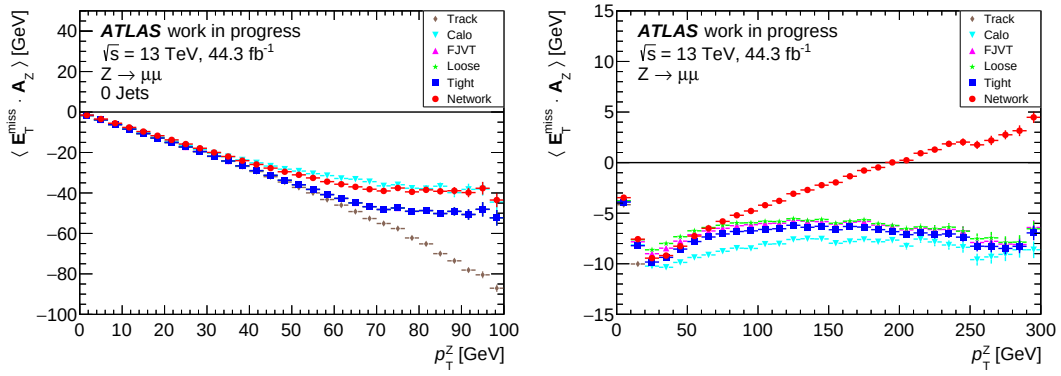
pileup, which is expected as the dominant terms are all track-based. However, Network $E_{\mathrm{T}}^{\mathrm{miss}}$ still showed a near consistent improvement over Tight of around 4 GeV across the full N_{PV} range, further supporting the fact that Network $E_{\mathrm{T}}^{\mathrm{miss}}$ was more accurately capturing the hadronic recoil.

10.1.3 Response

For studies of $E_{\mathrm{T}}^{\mathrm{miss}}$ response, Figure 10.7 shows $\langle \mathcal{P}_{||}^{Z} \rangle$ (defined in Equation 7.13) as a function of p_{T}^{Z} for the 0-jet and inclusive $Z \rightarrow \mu\mu$ sample, respectively. Performing this study in events with and without jets allowed the isolation of the jet and soft-term responses. In both plots, all $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms show the characteristic steep decrease in $\langle \mathcal{P}_{||}^{Z} \rangle$ with increasing p_{T}^{Z}, indicating an initial underestimation of the hadronic recoil. For events without jets in Figure 10.7(a), the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ shows better recoil response than the other pileup suppressed algorithms, and instead demonstrates a profile similar to Calo $E_{\mathrm{T}}^{\mathrm{miss}}$. This suggests that the neural network is accepting soft calorimeter signals not used in the Tight or Track algorithms to boost the reconstruction of the hadronic recoil. It is also able to do so without compromising its stability against pileup as shown in 10.6(b). The fact that the Network performs similarly to the track-based algorithms for $p_{\mathrm{T}}^{Z} < 25$ GeV in Figure 10.7(a) and for $\Sigma E_{\mathrm{T}} < 80$ GeV in Figure 10.6(a) suggests that it only begins to include contributions from the CST as the hardness of the event increases. It is showing the type of adaptive behaviour that it was designed for. However, even with these additional contributions, the neural network still underestimates the overall hadronic recoil without the presence of jets.

For the inclusive selection shown in Figure 10.7(b), all algorithms except Track $E_{\mathrm{T}}^{\mathrm{miss}}$ recover slightly after the initial decrease in $\langle \mathcal{P}_{||}^{Z} \rangle$. This indicates that more of the hadronic recoil was identified as a fully calibrated jet and thus was better represented in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. For the Calo, FJVT, Loose and Tight algorithms this

more complete representation of the hadronic recoil still possesses a persistent negative bias, with a residual offset of around 8 GeV for $p_{\mathrm{T}}^{Z} > 50$ GeV. The cause for this offset was explored in Reference [10] and was determined to be an underestimation of the soft-term in each algorithm, even Calo $E_{\mathrm{T}}^{\mathrm{miss}}$. Conversely, Network $E_{\mathrm{T}}^{\mathrm{miss}}$ seems to overestimate the response of the hadronic recoil and the values of $\langle \mathcal{P}_{||}^{Z} \rangle$ become positive in the range $p_{\mathrm{T}}^{Z} > 190$ GeV. Since this does not occur in the 0-jet sample, it indicates that the network is attempting to compensate for the undervalued soft terms of the standard $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithms by increasing the contribution of the reconstructed jets in the event. Unfortunately there were not enough statistics in the region $p_{\mathrm{T}}^{Z} > 300$ GeV to be able to determine the nature of this overestimation beyond this point.

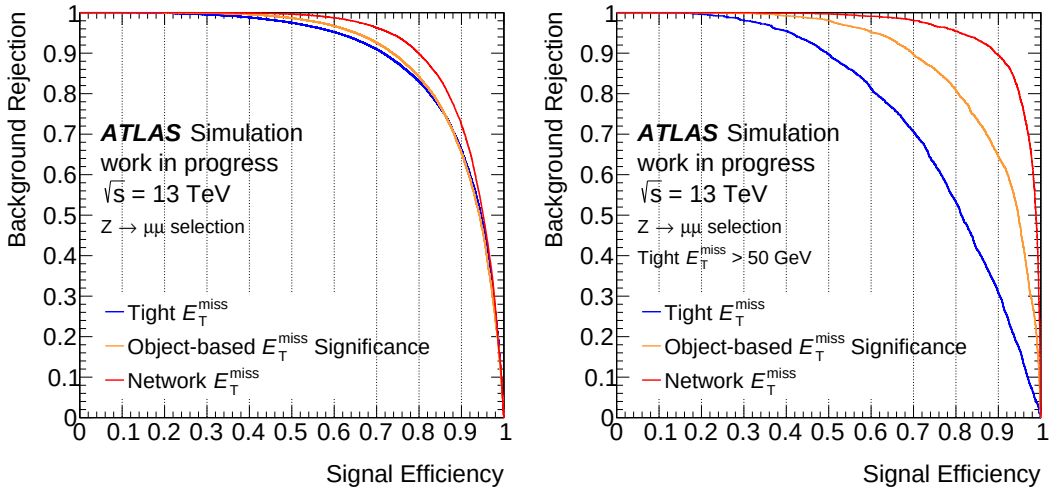


(a) (b)

Figure 10.7: Plots showing $\langle \mathcal{P}_{||}^{Z} \rangle = \langle \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} \cdot \mathbf{A}_{Z} \rangle$ as a function of p_{T}^{Z} using six $E_{\mathrm{T}}^{\mathrm{miss}}$ working points in a (a) 0-jet and (b) inclusive selection of $Z \to \mu\mu$ events extracted from data.

10.1.4 Separation Power

The object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance $\mathcal{S}_O$ was found to be a better discriminator between simulated $Z \to ee$ and $ZZ \to eevv$ events than $E_{\mathrm{T}}^{\mathrm{miss}}$ [224], as shown in Figure 7.1. So to estimate the potential gain of the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ working point, its separation power was similarly compared in the muon channel. Figure 10.8(a) shows that Network $E_{\mathrm{T}}^{\mathrm{miss}}$ offers a small but noticeable gain in separation power compared to the other two variables, Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ and $\mathcal{S}_O$ for the inclusive selection. In a subset where Tight $E_{\mathrm{T}}^{\mathrm{miss}} > 50$, shown in Figure 10.8(b), a much more substantial improvement is observed. Choosing a particular operating point at 90% signal efficiency: Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ provides around 32% background rejection, $\mathcal{S}_O$ gives 65%, and Network $E_{\mathrm{T}}^{\mathrm{miss}}$ gives 90% background rejection. This shows that the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ can be beneficial for event selection as it is able to distinguish events with fake $E_{\mathrm{T}}^{\mathrm{miss}}$ from events with genuine $E_{\mathrm{T}}^{\mathrm{miss}}$ more effectively. Recall that it is able to do this without having trained on events with no True $E_{\mathrm{T}}^{\mathrm{miss}}$.



(a) (b)

Figure 10.8: Receiver operating characteristic curves showing the background ($Z \to \mu\mu$) rejection versus signal ($ZZ \to \mu\mu\nu\nu$) efficiency in simulated samples using a $Z \to \mu\mu$ selection. The performance is shown using Tight $E_{\mathrm{T}}^{\mathrm{miss}}$, object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance ($\mathcal{S}_O$) and Network $E_{\mathrm{T}}^{\mathrm{miss}}$ as discriminants in (a) all events and in (b) events with Tight $E_{\mathrm{T}}^{\mathrm{miss}} > 50\,\mathrm{GeV}$. Figure (b) shows similar variables as Figure 7.1 with different data.

10.2 $E_{\mathrm{T}}^{\mathrm{miss}}$ in Final States With Neutrinos

Measuring $E_{\mathrm{T}}^{\mathrm{miss}}$ performance in final states without True $E_{\mathrm{T}}^{\mathrm{miss}}$ is useful because it allows one to study the reconstruction properties in data. However, it is not an ideal environment for testing a new algorithm if the inner workings of that algorithm is unknown. Neural networks are often referred to as black box estimators, and it would be impossible to fully understand how the one used in this analysis arrives at its output. If the network consistently underestimates the magnitude of True $E_{\mathrm{T}}^{\mathrm{miss}}$ or simply outputs zero, it would seemingly produce results with very high accuracy. So for a more complete understanding of the performance of Network $E_{\mathrm{T}}^{\mathrm{miss}}$, studies were conducted on final states that contained neutrinos and therefore did not have trivial target values of True $E_{\mathrm{T}}^{\mathrm{miss}} = 0$. However, this required that the studies could only be performed in MC simulation.

Three different selections were used in this evaluation, and the absence of background allowed for fairly loose selection criteria.

$t\bar{t}$ event selection

The $t\bar{t}$ process with non-all-hadronic decays allow for the evaluation of $E_{\mathrm{T}}^{\mathrm{miss}}$ performance in final states with high jet multiplicities. This process is also usually responsible for a significant portion of background in QCD sensitive BSM searches.

The event selection criteria were:

- The event contained at least four signal jets.
- At least one of the jets was b-tagged.

$WW \to l\nu l\nu$ event selection

This event topology is useful for studying the $E_{\mathrm{T}}^{\mathrm{miss}}$ response and resolution where the expected jet multiplicity is relatively low.

The following event selection was applied:

- The event contained exactly two opposite sign signal leptons, with no other baseline leptons present.
- The lowest unprescaled single lepton (electron or muon) trigger was fired and at least one of the leptons matched the trigger.
- Both the leading and the sub-leading lepton had $p_{\mathrm{T}} > 25\,\mathrm{GeV}$.

(VBF) $H \to WW$ event selection

The (VBF) $H \to WW$ event topology can be used to study the performance of $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction in events with forward jets. For this analysis, this signal region is particularly important as it involves a SM process not seen by the neural network during its training. The size of this dataset was notably smaller than all the others. Which is why there is a large amount of statistical variance present in the corresponding plots in the following sections.

The applied event selection was identical to the $WW \to l\nu l\nu$ region.

10.2.1 Resolution

The resolution was measured for final states with True $E_{\mathrm{T}}^{\mathrm{miss}} > 0$ using RMSE according to Equation 7.15 in the three aforementioned simulated samples. For these final states the resolution can be determined as a function of True $E_{\mathrm{T}}^{\mathrm{miss}}$ in addition to ΣE_{T}, as shown by Figure 10.9. For all samples and for all bins, Network $E_{\mathrm{T}}^{\mathrm{miss}}$ produced the best resolution compared to the other working points.

Figures 10.9(a) and 10.9(b) show the $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution in the simulated $t\bar{t}$ sample. With its high jet activity, this sample has notably worse resolution than the other final states regardless of working point. It also shows the greatest resolution difference between the Network and Tight working point, the next best performing algorithm. This shows that the network is mainly correcting for the measured jet momenta. Most of the performance gain is observed in the region of True $E_{\mathrm{T}}^{\mathrm{miss}} < 150\,\mathrm{GeV}$, beyond which little difference exists between Network $E_{\mathrm{T}}^{\mathrm{miss}}$ and the methods using the TST. This property is also visible in the diboson and Higgs samples, shown in

Figure 10.9(c) and Figure 10.9(f), respectively. Although for these final states, the threshold for when the resolution difference between the Tight and the Network algorithms becomes insubstantial is around $E_{\mathrm{T}}^{\mathrm{miss}} = 100\,\mathrm{GeV}$. This is investigated in Section 10.3.1.

The resolution was also plotted as a function of pileup measured by both N_{PV} and μ. Since the results for all three signal regions looked very similar, the plots for the $t\bar{t}$ sample are shown in Figure 10.10 while the plots for the other two processes are in Appendix A in Figure A.7. For the $t\bar{t}$ final state, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ has considerably better resolution than the other methods. Network $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution increases from around 24 GeV at $N_{\mathrm{PV}} = 20$ to around 26 GeV at $N_{\mathrm{PV}} = 40$. For the Tight working point, $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution increases from around 29 GeV at $N_{\mathrm{PV}} = 20$ to around 35 GeV at $N_{\mathrm{PV}} = 40$. While not independent of pileup, Network $E_{\mathrm{T}}^{\mathrm{miss}}$ is more resistant to the effects of additional pp interactions than any of the object-based algorithms. Therefore, the performance gain of the network over these methods is predicted to continue to increase for higher luminosity environments. Deviations from linearity for RMSE as a function of N_{PV} are due to an increase in vertex-merging [230] as pileup increases, explained in Section 9.4.

10.2.2 Response

The $E_{\mathrm{T}}^{\mathrm{miss}}$ response for final states with neutrinos is measured using the relative deviation from linearity Δ_T^{lin}, as defined in Equation 7.12. This was studied in all three signal regions. The results for the $t\bar{t}$ sample are shown in Figure 10.11, while the plots for the other two processes are shown in Figure A.8. The results show that the various algorithms overestimate the magnitude of $E_{\mathrm{T}}^{\mathrm{miss}}$ when True $E_{\mathrm{T}}^{\mathrm{miss}}$ is close to zero. This is due to the positive observation bias inherent in $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction. The Network $E_{\mathrm{T}}^{\mathrm{miss}}$ shows the steepest descent in Δ_T^{lin} indicating that it substantially reduces the bias. Since the network was observed to provide the most performance gain when True $E_{\mathrm{T}}^{\mathrm{miss}} < 150\,\mathrm{GeV}$, Figure 10.9(a), it seems that correcting for the observation bias is one of the main strengths of the network. However, in the range True $E_{\mathrm{T}}^{\mathrm{miss}} > 150\,\mathrm{GeV}$, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ shows a loss of response by around 10%, whereas all other object-based algorithms demonstrate good linearity. So while Network $E_{\mathrm{T}}^{\mathrm{miss}}$ seems to produce a better overall resolution, it is inducing a negative bias. The only other working point to show a systematic underestimation of the magnitude was Track $E_{\mathrm{T}}^{\mathrm{miss}}$, as this algorithm does not include jet contributions. As discussed in Section 10.1.3, the neural network seems to be amplifying the jet response, not ignoring it. Therefore, the reason for the negative bias in Network $E_{\mathrm{T}}^{\mathrm{miss}}$ is probably from another source. This is discussed in Section 10.3.

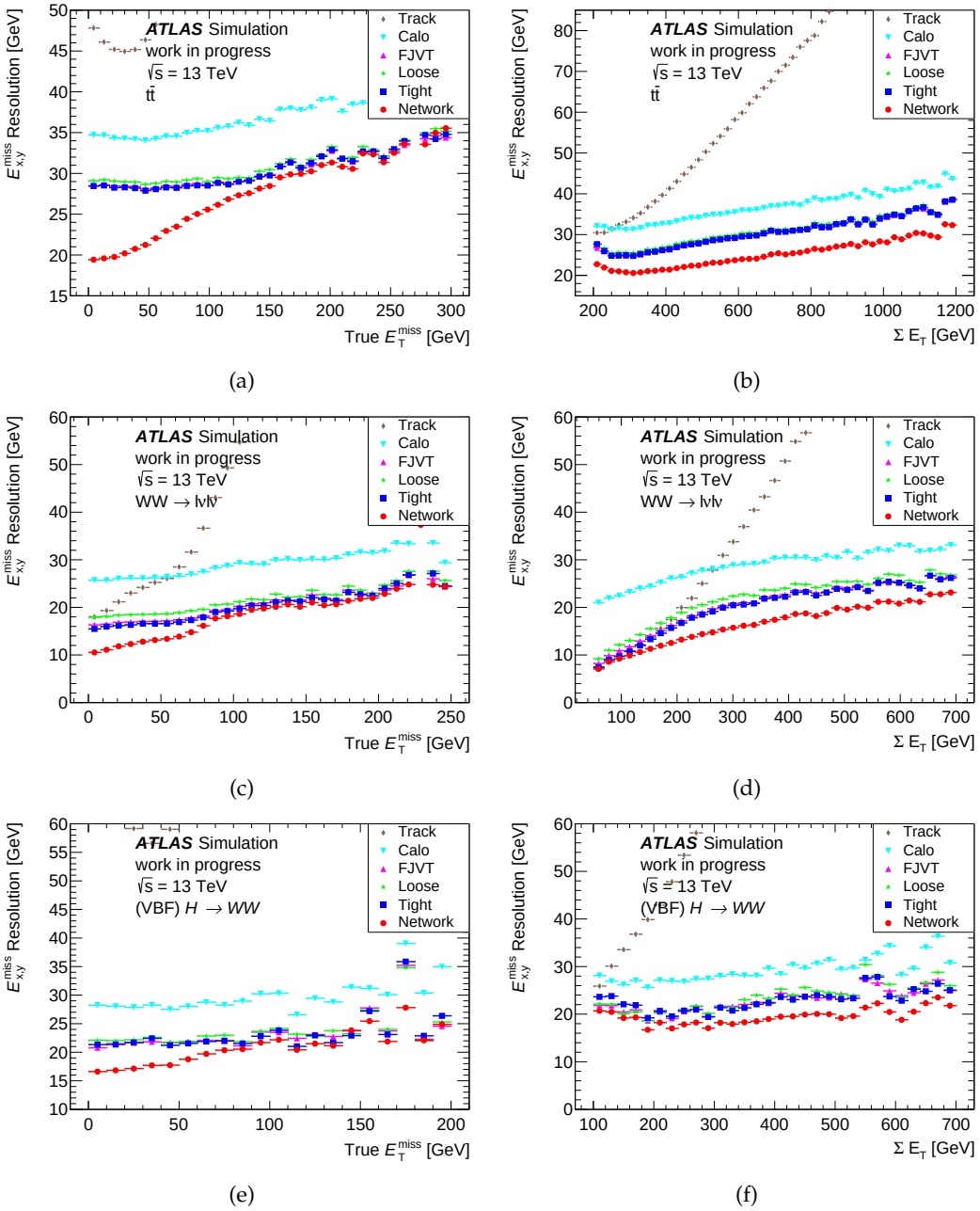
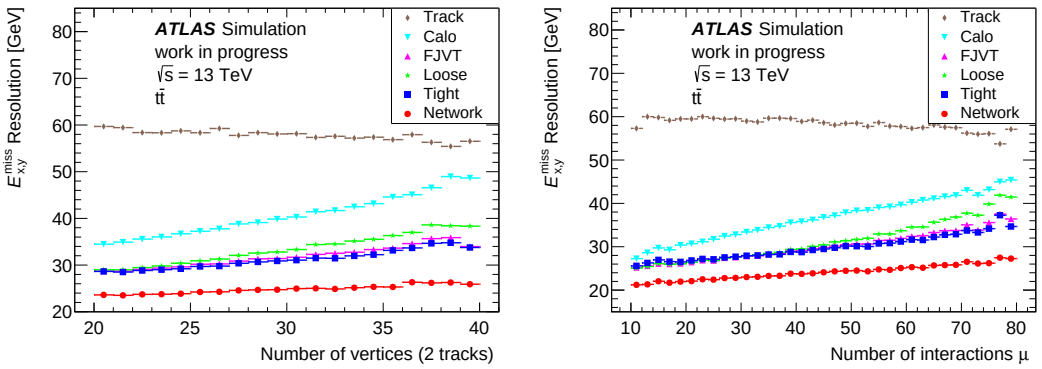


Figure 10.9: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions measured by RMSE using six different working points in a MC $t\bar{t}$ sample plotted versus (a) True $E_{\mathrm{T}}^{\mathrm{miss}}$ and (b) Tight ΣE_{T} . The same profiles in a $WW \rightarrow l\nu l\nu$ sample are shown in (c) and (d), and again in a (VBF) Higgs sample in (e) and (f).



(a) (b)

Figure 10.10: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions measured by RMSE using six different working points in a MC $t\bar{t}$ sample are shown versus pileup measured by (a) N_{PV} and (b) mu.

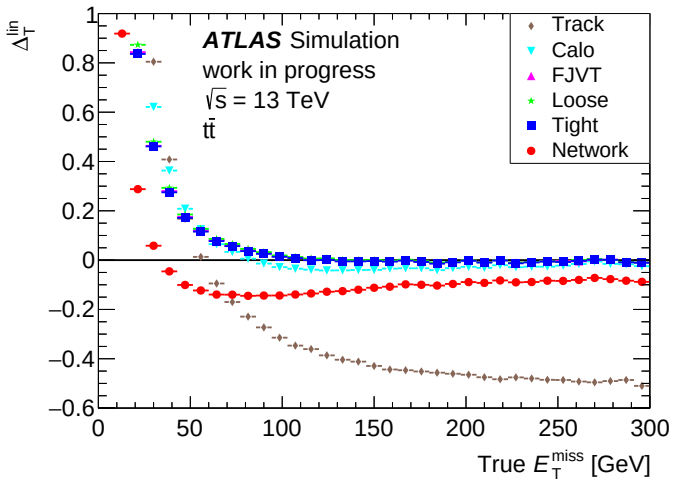


Figure 10.11: The deviation of the $E_{\mathrm{T}}^{\mathrm{miss}}$ response from linearity using six different working points measured as a function of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ in $t\bar{t}$ final states in MC simulations.

10.2.3 Angular Resolution

The angular resolution described in Section 7.2.3 was measured in all regions. It is plotted as a function of True $E_{\mathrm{T}}^{\mathrm{miss}}$ for the $t\bar{t}$ final state in Figure 10.12, and for the $WW \rightarrow lvlv$ and Higgs samples in Figure A.9. The Network $E_{\mathrm{T}}^{\mathrm{miss}}$ offers the best angular resolution, slightly but consistently improving on the estimates of the Tight working point for all values of True $E_{\mathrm{T}}^{\mathrm{miss}}$.

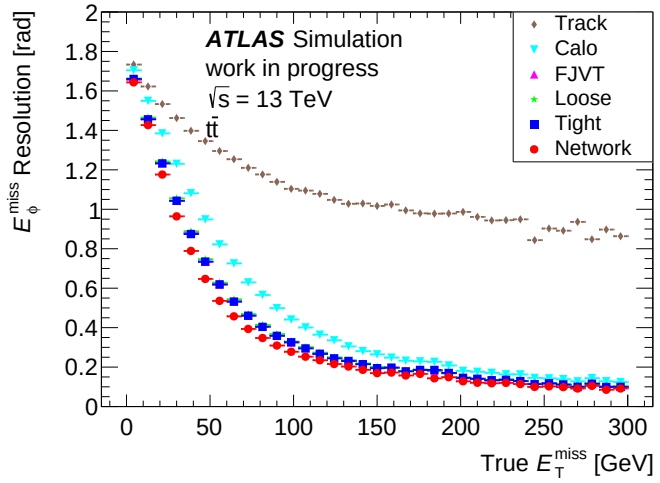


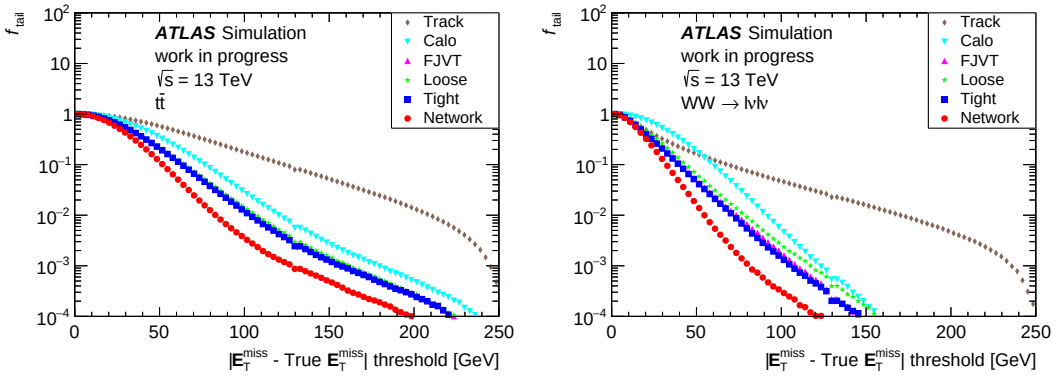
Figure 10.12: The angular resolution measured by the RMSE of the reconstructed ϕ^{miss} distribution plotted in bins of True $E_{\mathrm{T}}^{\mathrm{miss}}$ for six working points in a simulated $t\bar{t}$ sample.

10.2.4 Distribution Tails

The tail fraction f_{tail} as defined in Section 7.2.4 was calculated using a range of $|\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} - \text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|$ thresholds in the $t\bar{t}$ and $WW \rightarrow lvlv$ samples. The results are shown in Figure 10.13. The tails for all working points are larger for the $t\bar{t}$ sample because of the enhanced jet response and multiplicity. The Network $E_{\mathrm{T}}^{\mathrm{miss}}$ produces the steepest decrease in both tail distributions compared to the other methods. This shows that not only is the neural network more accurately reconstructing the $E_{\mathrm{T}}^{\mathrm{miss}}$ for a majority of the events, but also that it is not producing noticeable tails or outliers.

10.3 Dependence of the Performance on the Training Set

The previous section discussed results that showed that the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ provided the best resolution for final states that contained neutrinos. The most improvement was observed when True $E_{\mathrm{T}}^{\mathrm{miss}}$ was relatively small. However, it was also observed that on average, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ underestimated the magnitude when True $E_{\mathrm{T}}^{\mathrm{miss}} > 40$, as Shown in Figure 10.11.



(a) (b)

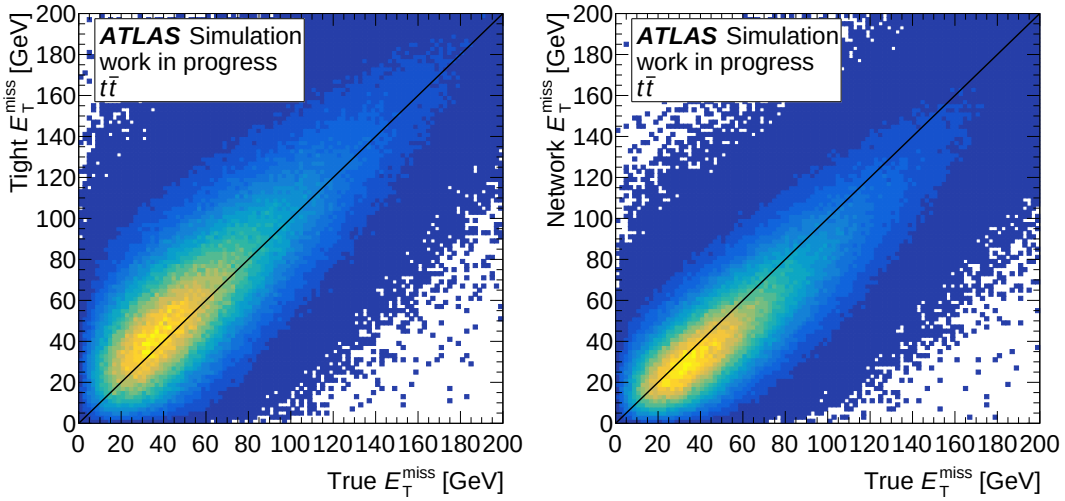
Figure 10.13: The tail fraction using a threshold applied to $|\mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} - \text{True } \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}}|$, the Euclidean distance between the true missing transverse momentum and the reconstructed vector, using six different $E_{\mathrm{T}}^{\mathrm{miss}}$ working points in a sample of simulated (a) $t\bar{t}$ and (b) $WW \to l\nu l\nu$ final states.

In addition to the deviation from linearity plots, the response of the Tight and Network working points in the $t\bar{t}$ sample are shown in Figure 10.14. These 2D histograms plot the True $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitude on the x-axis and the reconstructed magnitude on the y-axis. Therefore, any point above the $y = x$ line indicates an event whose magnitude was overestimated. Conversely, any point below the line indicates an event whose magnitude was underestimated by the specific algorithm. Figure 10.14(a), which plots the Tight working point, shows a roughly even spread above and below the diagonal when True $E_{\mathrm{T}}^{\mathrm{miss}} > 100\,\mathrm{GeV}$, corresponding to the range where Tight Δ_T^{lin} was approximately zero in Figure 10.11. The majority of events in the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution in Figure 10.14(b) lie below the diagonal.

The neural network's negative bias in $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitude seems to be caused by the shape of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution in its training set. This distribution is shown in Figure 9.3[†]. The vast majority of events used directly to train the neural network lie in the range $10\,\mathrm{GeV} < \text{True } E_{\mathrm{T}}^{\mathrm{miss}} < 100\,\mathrm{GeV}$. The peak of the distribution is found at around True $E_{\mathrm{T}}^{\mathrm{miss}} = 40\,\mathrm{GeV}$, which also happens to be the value where Network Δ_T^{lin} intersects with the x-axis in Figure 10.11. This suggests that the neural network tends to shift its predictions towards values closer to this modal range.

This is because the neural network learns by minimising the average cost over the entire training set. A simple, initial step in this process is to only output values within the bulk of the target distribution. As learning progresses, the network moves on to develop more non-trivial methods and predictions. Since the Network was shown to improve the resolution across the full True $E_{\mathrm{T}}^{\mathrm{miss}}$ range, it did develop these complex methods, but it seemingly never outgrew its initial bias.

[†]The figure shows the True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution of the Learning class. However, the training set used for the final model was simply the Learning class after 5% events randomly removed, so the distribution shape is the same.



(a) (b)

Figure 10.14: Heat-maps showing the distributions of a MC $t\bar{t}$ sample where the True $E_{\mathrm{T}}^{\mathrm{miss}}$ is plotted along the x-axis and the reconstructed $E_{\mathrm{T}}^{\mathrm{miss}}$ is plotted along the y-axis using the (a) Tight and the (b) Network working points, respectively.

Given enough training, there is no obvious reason why the neural network did not eventually learn to estimate the magnitude in the tail correctly. But it is possible that the network began to overfit before this could take place. So, effort was made to further adjust the network architecture to combat overfitting. This included applying more aggressive regularisors and a reduction in network size. The depth and optimiser of the network were also changed, as some studies claim that wider networks equipped with the Adam algorithm tend to be more prone to overfitting and that deeper networks using standard gradient descent with momentum produce the best testing performance [140]. The loss function was changed from Huber to L2, since L2-loss is more likely to produce an unbiased estimator [75]. However, it is important to note that the network was predicting True $E_{\parallel}^{\mathrm{miss}}$ and True $E_{\perp}^{\mathrm{miss}}$ - not the magnitude directly - and because of the standardisation step discussed in Section 9.4.5, the means of these distributions were zero. Therefore, an unbiased estimator of these target values would not necessarily also be an unbiased estimator of the magnitude. So, further networks were trained using only the True $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitude as the target value. None of these changes or redesigns resulted in a network that did not possess the same systematic loss of response.

So, the problem of the uneven training set was then tackled directly. When training neural networks for classification, sometimes the groups of labelled data are imbalanced. One of the commonly used tactics is to over sample the underrepresented classes during each training epoch, artificially balancing out the targets [76]. The same philosophy was applied in the context of this regression problem, where oversampling was used to improve the contribution of events in the tail of the True $E_{\mathrm{T}}^{\mathrm{miss}}$. Ideally, the sampling weight for each event would be inversely proportional to the value of the probability distribution function (pdf) of True $E_{\mathrm{T}}^{\mathrm{miss}}$ in the training set.

Since that pdf has no clear analytical shape, this process was approximated by using a histogram with 100 bins in the range 0 to 300 GeV. For each event within a bin, the same weight was attributed which was equal to the inverse of the bin height, thereby approximately flattening the True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution up to 300 GeV. The overall distribution was then normalised to match the initial integral before the weights were applied.

An adverse effect of this approach was that the effective size and diversity of the training set actually decreased. This was because the network was treating the set of 2326 events in the range 297 GeV $<$ True $E_{\mathrm{T}}^{\mathrm{miss}}$ $<$ 300 GeV with the same amount of importance as the 353 617 events in the range 36 GeV $<$ True $E_{\mathrm{T}}^{\mathrm{miss}}$ $<$ 39 GeV. While this was the intended effect, it did mean that less events had more say in how the network learned. This led to more unstable descent and the network began to overfit much earlier in its training process. Since the multiplicity of events approaches zero as True $E_{\mathrm{T}}^{\mathrm{miss}}$ gets large, flattening the entire distribution would require weights that approach infinity. All events in the training set beyond the limits of this histogram received the same weight as those in the final bin.

Creating a sampling technique that was fast enough to not affect the training times severely was very challenging, even with built in libraries for bootstrapping. Instead, the events were sampled as before, but the weights were directly applied to the loss function. Usually, the learning algorithm makes a single iteration to minimise the average loss calculated over a mini-batch. This was modified to a weighted average of the loss, as shown by Equation 10.1. From the perspective of the descent algorithm, this was mathematically equivalent to a weighted random selection of the mini-batch given an infinite number of draws with replacement, but much faster.

$$\text{Standard mini-batch cost:} \qquad C_M(\boldsymbol{\theta}) = \frac{1}{M}\sum_{i=1}^{M} L(\hat{\mathbf{y}}_i, \mathbf{y}_i) \tag{10.1}$$

$$\text{Modified mini-batch cost:} \qquad \tilde{C}_M(\boldsymbol{\theta}) = \frac{1}{\sum w_i}\sum_{i=1}^{M} w_i L(\hat{\mathbf{y}}_i, \mathbf{y}_i) \tag{10.2}$$

10.3.1 Performance of Networks Trained on Different True $\boldsymbol{E}_{\mathbf{T}}^{\mathbf{miss}}$ Distributions

The neural network trained using a pseudo-flat True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution was given the label (A). The (A) Network $E_{\mathrm{T}}^{\mathrm{miss}}$ did return an unbiased response for events with True $E_{\mathrm{T}}^{\mathrm{miss}}$ $>$ 70 GeV in the MC $t\bar{t}$ sample, as shown by Figure 10.15(a) and Figure 10.17. However, it also resulted in an underestimation of the magnitude for many events in the range 10 GeV $<$ True $E_{\mathrm{T}}^{\mathrm{miss}}$ $<$ 40 GeV. This is shown by the large collection of events close to the x-axis in Figure 10.15(a). This resulted in considerably

worse overall resolution compared to the original model, and a massive reduction in the model's separation power between events with genuine and fake $E_{\mathrm{T}}^{\mathrm{miss}}$.

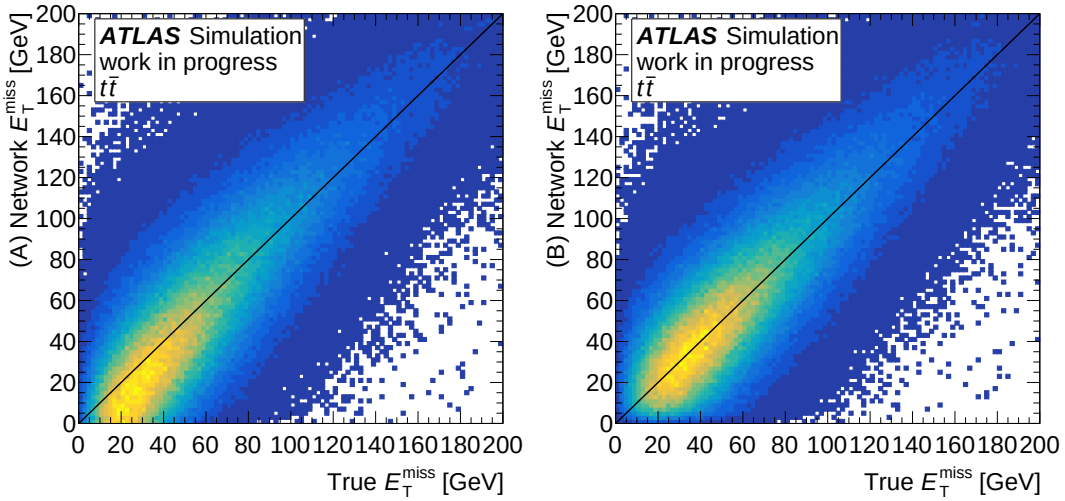


Figure 10.15: Heat-maps showing the distributions of a MC $t\bar{t}$ sample where the True $E_{\mathrm{T}}^{\mathrm{miss}}$ is plotted along the x-axis and the neural network reconstructed $E_{\mathrm{T}}^{\mathrm{miss}}$ is plotted along the y-axis. The network in (a) was trained on a set that was approximately flat in True $E_{\mathrm{T}}^{\mathrm{miss}}$. The network in (b) was trained on a set that was approximately flat in True $E_{\mathrm{T}}^{\mathrm{miss}}$ only after the peak at 40 GeV.

A hypothesis for why the (A) Network severely underestimated the True $E_{\mathrm{T}}^{\mathrm{miss}}$ in the low region is based on the following arguments. First, is that the standard $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction methods have a finite and limited resolution, so when the true value of the measurand is smaller - or of equivalent scale - to the resolution, these methods tend to overestimate. This gives rise to the previously discussed positive observation bias. For events in this range, a model that trivially outputs zero would be recorded as having superior resolution to all other methods, as shown by Figure 10.16. A neural network may learn that when the estimated $E_{\mathrm{T}}^{\mathrm{miss}}$ of some of the working points is below a certain value, it would be more numerically favourable to simply output a value close to zero. This happened in the (A) Network because by flattening out the training set, the contribution of events with True $E_{\mathrm{T}}^{\mathrm{miss}}$ close to zero was inflated. This behaviour should be discouraged if the model is to be used as an effective separator of events with genuine and fake $E_{\mathrm{T}}^{\mathrm{miss}}$.

In response to these findings, an additional sampling technique was attempted in order to fix the loss of response found in the original model, without producing a network that would overzealously estimate $E_{\mathrm{T}}^{\mathrm{miss}} = 0$. The location of the peak of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution in the training set (Figure 9.3) was found to correspond to the lower limit of the range in which the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ exhibited its negative bias (Figure 10.11); which was around 40 GeV. This sampling technique followed a similar method to the one before, but only the bins with True $E_{\mathrm{T}}^{\mathrm{miss}} > 40$ GeV were upscaled

to match the height of the modal bin. While this method sought to address the problems discussed above, it is not underpinned by solid statistical theory. But in the field of machine learning, empirical performance is more important than strong theoretical motivation. This network, labelled (B), not only led to good linear response, but it also did not underestimate small values of True $E_{\mathrm{T}}^{\mathrm{miss}}$, thus fixing both problems as shown by Figure 10.15(b) and Figure 10.17. However, this sampling method still caused early overfitting and unstable learning due to the aforementioned decrease in effective training set size associated with oversampling. Its resolution was better than the original model only when True $E_{\mathrm{T}}^{\mathrm{miss}} > 115\,\mathrm{GeV}$, as shown by Figure 10.18(a), but for the sample as a whole, it was 15% worse, Figure 10.18(b). In summary, there exists a trade-off with performance at high and low True $E_{\mathrm{T}}^{\mathrm{miss}}$ due to the differing amount of available statistics.

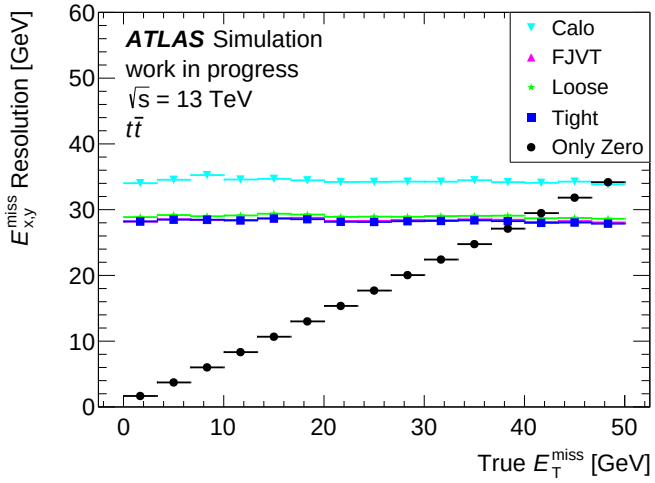


Figure 10.16: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions of four working points in bins of True $E_{\mathrm{T}}^{\mathrm{miss}}$. Due to the positive observation bias exhibited in the reconstruction algorithms, when the measurand is smaller or similar in scale to the resolutions, the working points are seemingly outperformed by a model that simply estimates $E_{\mathrm{T}}^{\mathrm{miss}} = 0$.

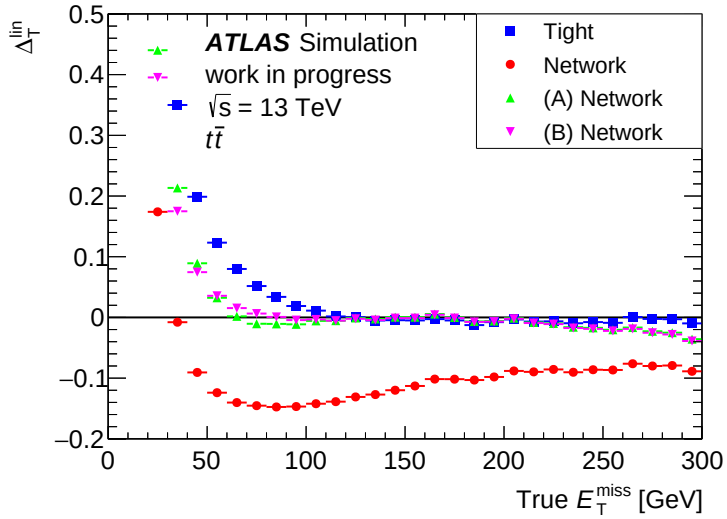
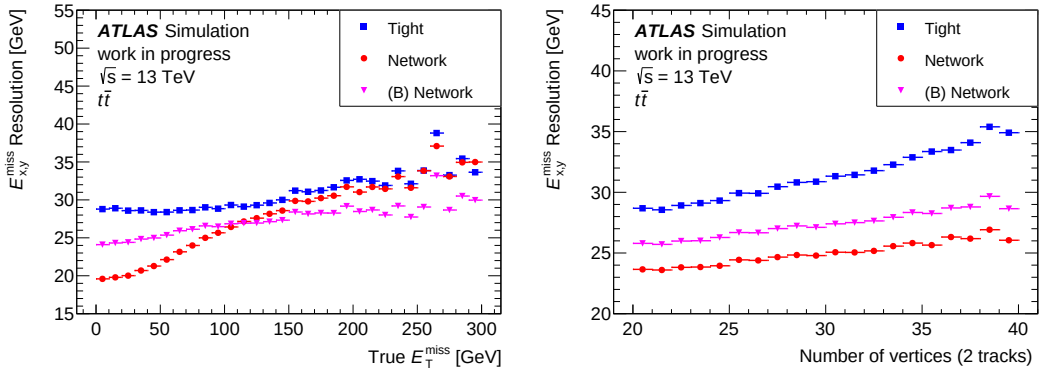


Figure 10.17: The deviation of the E_T^{miss} response from linearity using the Tight working point and three neural networks measured as a function of the True E_T^{miss} in $t\bar{t}$ final states in MC simulations.



(a) (b)

Figure 10.18: The E_T^{miss} resolutions measured by RMSE using the Tight working point and two neural networks in a MC $t\bar{t}$ sample plotted as a function of (a) True E_T^{miss} and (b) pileup. The (B) Network E_T^{miss} working point corresponds to the one trained on a set where events were over sampled if they had True $E_T^{miss} > 40\,\text{GeV}$.

10.3.2 Discussion in Response vs Resolution

When presenting this work to members of the ATLAS $E_{\mathrm{T}}^{\mathrm{miss}}$ performance group, they expressed that for most analyses $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution is significantly more important than response. Therefore, the increase in linearity achieved when using the oversampling techniques was deemed to be not worth the detriment to the overall resolution. So despite its negative bias, the original model was used to create the results presented in the following chapter. But this study still served to demonstrate how alternative sampling of the training set could result in models with various performance in different regions of True $E_{\mathrm{T}}^{\mathrm{miss}}$. The original model was chosen due to its more consistent performance thanks to its better statistics. Its higher accuracy in the range True $E_{\mathrm{T}}^{\mathrm{miss}} < 115\,\mathrm{GeV}$ would mean that less events would make it into the signal regions of the following analysis due to large amounts of fake $E_{\mathrm{T}}^{\mathrm{miss}}$.

One of the suggestions received by the ATLAS $E_{\mathrm{T}}^{\mathrm{miss}}$ performance group would be to add certain SUSY samples to the training set, as these samples could be generated with True $E_{\mathrm{T}}^{\mathrm{miss}}$ much greater than 300 GeV. Therefore, the network could be trained on a set which is flatter, but without the adverse effects of oversampling. This is listed as a possible avenue of future work in Section 12.1.

Chapter 11

Performance Gain of Network $E_{\mathrm{T}}^{\mathrm{miss}}$ in a SUSY Search

SUSY, introduced in Section 2.2, is one of the most studied extensions of the SM. However, despite significant searches for evidence over the past decade [73], at this time no experimental results have confirmed the theory. In R-parity conserving models, the LSP - typically thought to be the lightest neutralino $\tilde{\chi}_1^0$ - is stable, and if it was produced in a pp collision in ATLAS it would escape the system undetected. This would lead to an increase in the number of recorded events with large $E_{\mathrm{T}}^{\mathrm{miss}}$, beyond what would be expected from SM processes alone. This characteristic is exploited in many SUSY searches at ATLAS, and is why the measurement of $E_{\mathrm{T}}^{\mathrm{miss}}$ with high accuracy is important in these studies.

This chapter serves to demonstrate the potential performance gain of using the neural network for $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction in a typical search for SUSY signals. This analysis replicates a previous study conducted by ATLAS [222] which looked for evidence of the electroweak production of charginos ($\tilde{\chi}_1^{\pm}$), neutralinos ($\tilde{\chi}_1^0$) and sleptons ($\tilde{l}$), which were described in Section 2.2. The information provided in this chapter is only a brief summary, and further details can be obtained from the original paper. Any changes from the original method are discussed. One such difference is that the original paper used 36.1 fb^{-1} of data recorded in 2016, while this replicated study used data recorded by ATLAS in 2017 with a total integrated luminosity of 43 fb^{-1}.

The original study was replicated twice, first using Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ and then using Network $E_{\mathrm{T}}^{\mathrm{miss}}$. These are referred to as the Tight study and the Network study, respectively. This chapter focuses on how the better resolution of the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ decreased the contribution of SM backgrounds in the signal region (SR) while still retaining similar signal efficiencies. While data is present in these studies it is mainly used to normalise the control regions (CRs), validate the results, and show that the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ does not cause disagreement between the data and MC. The main results are the MC derived signal sensitivities $\mathcal{S}_S$, calculated using the signal-over-root-background (SorB) which a common metric for detection sensitivity.

The original paper looked at multiple SRs, including final states with two leptons (with and without jets) and final states with three leptons. In this dissertation, only the investigation in a single region, which required two same-flavour leptons and no jets (2L-0J-SF)*, is repeated to demonstrate the benefits of Network $E_{\mathrm{T}}^{\mathrm{miss}}$. The preselection for this region includes:

- The event contained exactly two same-flavour opposite-sign SFOS signal leptons (muons or electrons), with no other baseline leptons present.
- Both leptons had $p_{\mathrm{T}} > 25\,\mathrm{GeV}$.
- The lowest unprescaled single lepton trigger was fired and at least one of the leptons matched the trigger.
- The dilepton invariant mass (m_{ll}) was greater than 60 GeV.
- The event contained zero non-b-tagged jets with $p_{\mathrm{T}} > 60\,\mathrm{GeV}$.

This region is sensitive to $\tilde{\chi}_1^{\pm}\tilde{\chi}_1^{\mp}$ and $\tilde{l}\tilde{l}$ pair production as shown in Figures 11.1(a) and 11.1(b).

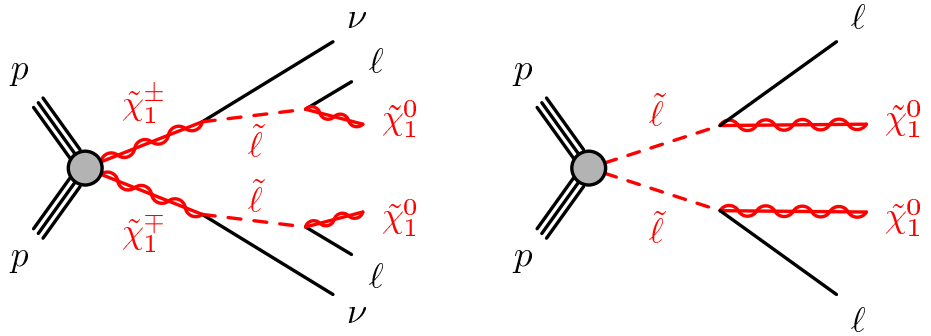


Figure 11.1: Feynman diagrams of the processes considered in this study [222]. For the production of $\tilde{\chi}_1^{\pm}\tilde{\chi}_1^{\mp}$ (a) it is assumed that the sleptons are light and thus accessible in the sparticle decay chains. For direct $\tilde{l}\tilde{l}$ production (b) each slepton decays into a lepton and a $\tilde{\chi}_1^0$ with a 100% branching ratio. All sleptons are assumed to be mass degenerate.

11.1 Stransverse Mass

An important variable used in this search is m_{T2}, also known as the stransverse mass [251, 252]. This variable is similar to the transverse mass (m_{T}), which is used for W boson mass (m_W) measurements in colliders [5]. Both observables provide estimates for the masses of particles produced in a collision where the longitudinal momentum of the hard scatter is unmeasured. The m_{T2} allows estimations of the masses of particles which were pair produced, decaying into a final state containing one or more invisible, but also massive particles.

*The name of this SR is taken directly from the original paper.

In final states with two leptons and some unseen particles, m_{T2} is defined by:

$$m_{T2} = \min_{\mathbf{q}_T} \left[\max \left(m_T(\mathbf{p}_T^{l1}, \mathbf{q}_T), m_T(\mathbf{p}_T^{l2}, \mathbf{E}_T^{miss} - \mathbf{q}_T) \right) \right], \tag{11.1}$$

where $\mathbf{p}_T^{l1}$ and $\mathbf{p}_T^{l2}$ are the transverse momentum vectors of the leptons, and $\mathbf{q}_T$ is the transverse momentum vector that minimises the expression. Because of this complex minimisation, m_{T2} must usually be calculated computationally. In this analysis this was done using resources described in Reference [253].

For SM backgrounds involving $t\bar{t}$ or WW production, if the E_T^{miss} and the pair of selected leptons originate from two $W \to l\nu$ decays and all momenta are accurately measured, then the m_{T2} must be less than m_W. This is shown in Figure 11.2 which plots the normalised distributions of m_{T2} using the generator level True E_T^{miss}, in several SM datasets after the 2L-0J-SF preselection. The processes $t\bar{t}$, Wt and $W\bar{t}$ showed very similar shapes and thus were combined into the single distribution labelled Top. The Top and WW distributions have visible trailing edges at $m_{T2} = m_W = 80.379\,\text{GeV}$. The few events beyond that edge can be attributed to misreconstruction in the lepton momenta, the production of additional neutrinos, or an off-shell mass W boson. The ZZ and WZ are not similarly bounded in m_{T2}. The overwhelming majority of $Z \to ll$ events have m_{T2} close to zero.

SUSY processes may result in much larger values of m_{T2}. For example, in the direct production of $\tilde{l}\tilde{l}$ as shown by Figure 11.1(b), m_{T2} can be shown to be bounded only by:

$$m_{T2} \leq \frac{(m_{\tilde{l}})^2 - (m_{\tilde{\chi}_1^0})^2}{(m_{\tilde{l}})}. \tag{11.2}$$

Therefore, requiring m_{T2} to be significantly larger than m_W strongly suppresses several SM processes while maintaining good efficiency for many SUSY signals. Detector inefficiencies, primarily in E_T^{miss} reconstruction, result in many SM processes possessing m_{T2} values above their kinematic thresholds. Therefore in this study, particular attention was given to the impact of the Network E_T^{miss} on the background contributions of $Z \to ll$, WW, $t\bar{t}$, Wt and $W\bar{t}$ in a SR with a high m_{T2} cut.

11.2 Signal Regions and SUSY Samples

In addition to the preselection, the inclusive SR labelled 2L-0J-SF contained the cuts shown in Table 11.2. Events were required to have $m_{T2} > 100\,\text{GeV}$, the m_{ll} was required to be greater than 111 GeV to reduce the contribution of Z events. To further suppress $t\bar{t}$ contributions, the events could not contain any b-tagged jets. The 2L-0J-SF inclusive SR is broken up into 13 orthogonal exclusive SRs based on bins in m_{T2} and m_{ll} to maximise exclusion sensitivity across the simplified model parameter space of $\tilde{\chi}_1^{\pm}\tilde{\chi}_1^{\mp}$ and $\tilde{l}\tilde{l}$ production. All exclusive signal regions are shown in Table 11.1.

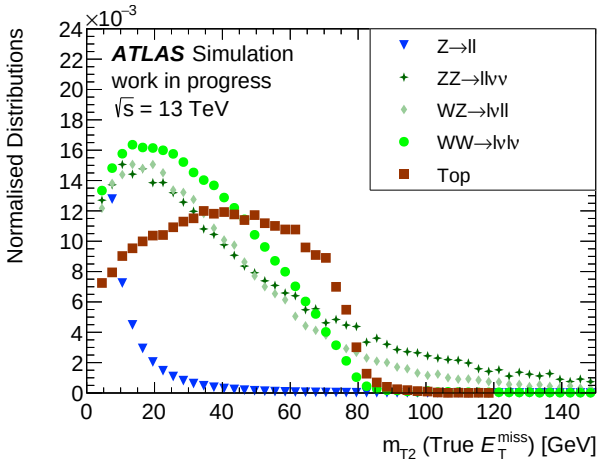


Figure 11.2: The m_{T2} distributions, constructed using generator level values of True E_T^{miss}, for various SM processes after the preselection of 2L-0J-SF was applied. The Top distribution is a combination of the $t\bar{t}$, Wt and $W\bar{t}$ samples, all of which had similar shapes.

2L-0J-SF exclusive signal region definitions		
m_{T2} [GeV]	m_{ll} [GeV]	SF bin
100-150	$111-150$	SF-a
	$150-200$	SF-b
	$200-300$	SF-c
	>300	SF-d
150-200	$111-150$	SF-e
	$150-200$	SF-f
	$200-300$	SF-g
	>300	SF-h
200-300	$111-150$	SF-i
	$150-200$	SF-j
	$200-300$	SF-k
	>300	SF-l
>300	>111	SF-m

Table 11.1: The definitions of the 13 exclusive signal regions which are defined by bins in both m_{T2} and m_{ll}.

Three simulated samples of direct $\tilde{l}\tilde{l}$ production (Section 8.2.2) were included, using different combinations of slepton and neutralino masses, $m(\tilde{l}, x_l^0)$. These mass combinations were $(300, 200)$ GeV, $(400, 250)$ GeV and $(550, 1)$ GeV. They correspond to upper limits of m_{T2} of around 167 GeV, 243 GeV and 550 GeV, respectively. These couplets were selected as they lie close to the exclusion regions based on the results of the original paper [222].

11.3 Background Estimation and Validation

The dominant SM backgrounds of the 2L-0J-SF signal region are irreducible processes from SM diboson events (WW, WZ, ZZ) and the dileptonic decays of top events ($t\bar{t}$, Wt, $W\bar{t}$). The expected kinematic distributions of these processes were

taken from MC simulations, and then normalised to data in two dedicated CRs labelled CR-VV and CR-top. Also expected was a small contribution from $Z \to ll$ events with large amounts of fake $E_{\mathrm{T}}^{\mathrm{miss}}$ and poorly measured lepton momenta; this was taken directly from MC. It is important to note that reducible backgrounds from fake or non-prompt leptons were not included, as the datasets needed to estimate these backgrounds were not available. The non-prompt events contributed very little to all regions in the original paper [222], so any difference due to their exclusion is expected to be small. The total background estimation was checked on a dedicated validation region (VR) labelled VR-VV.

The cuts for CR-VV, CR-top and VR-VV are shown in Table 11.2. CR-VV was kept orthogonal from 2L-0J-SF by requiring that $|\mathrm{m}_{ll} - \mathrm{m}_Z| < 20\,\mathrm{GeV}$. To reduce the contribution of $Z \to ll$ it also required that $\mathrm{m}_{\mathrm{T2}} > 130\,\mathrm{GeV}$ and $E_{\mathrm{T}}^{\mathrm{miss}} > 100\,\mathrm{GeV}$. This region was dominated by ZZ processes and subdominant in WZ and WW, but the same normalisation factor was applied to all three. For CR-top, the main distinguishing cut was that it required at least one b-tagged jet. In the original paper, these control regions were used in a simultaneous global fit, but in this project normalisation factors for the diboson and top events were estimated from their individual CRs. Since the purity of both of these CRs was over 99%, this change was expected to have had a minimal effect on the final outcome.

Region	**CR-VV**	**CR-top**	**VR-VV**	**2L-0J-SF**
Lepton pair	SFOS	SFOS	SFOS	SFOS
$n_{\text{non-}b\text{-tagged jets}}$	0	0	0	0
$n_{b\text{-tagged jets}}$	0	1	0	0
$\lvert\mathrm{m}_{ll} - \mathrm{m}_Z\rvert$ [GeV]	< 20	> 20	> 20	> 20
m_{T2} [GeV]	> 130	$75 - 100$	$75 - 100$	> 100
$E_{\mathrm{T}}^{\mathrm{miss}}$ [GeV]	> 100	-	-	-

Table 11.2: The definition of the two CRs, the VR and the inclusive SR. The p_{T} thresholds placed on the requirements for b-tagged and non-b-tagged jets correspond to 20 GeV and 60 GeV, respectively.

For the Tight study, the normalisation factors returned for the top and diboson backgrounds were 1.058 ± 0.045 and 1.232 ± 0.064 respectively. For the Network study, the corresponding values were 1.092 ± 0.055 and 1.239 ± 0.141. The plots in Figure 11.3 show the $E_{\mathrm{T}}^{\mathrm{miss}}$ and m_{T2} distributions for the data and estimated backgrounds in VR-VV with the normalisations applied. Similar plots for the two CRs are shown in Figures A.11 and A.12 in Appendix A. Overall good agreement is shown between data and MC for both the Network study and the Tight study. While ratio plots for the Network study do seem to possess more variance, the number of events was also substantially lower and the fluctuations are accounted for by statistical uncertainty. All confidence intervals only account for uncertainties due to: statistics, cross-section measurements, luminosity and (where appropriate) normalisation factors based on the CRs, only. An added effect of this is that the uncertainties vanished when the event counts were zero.

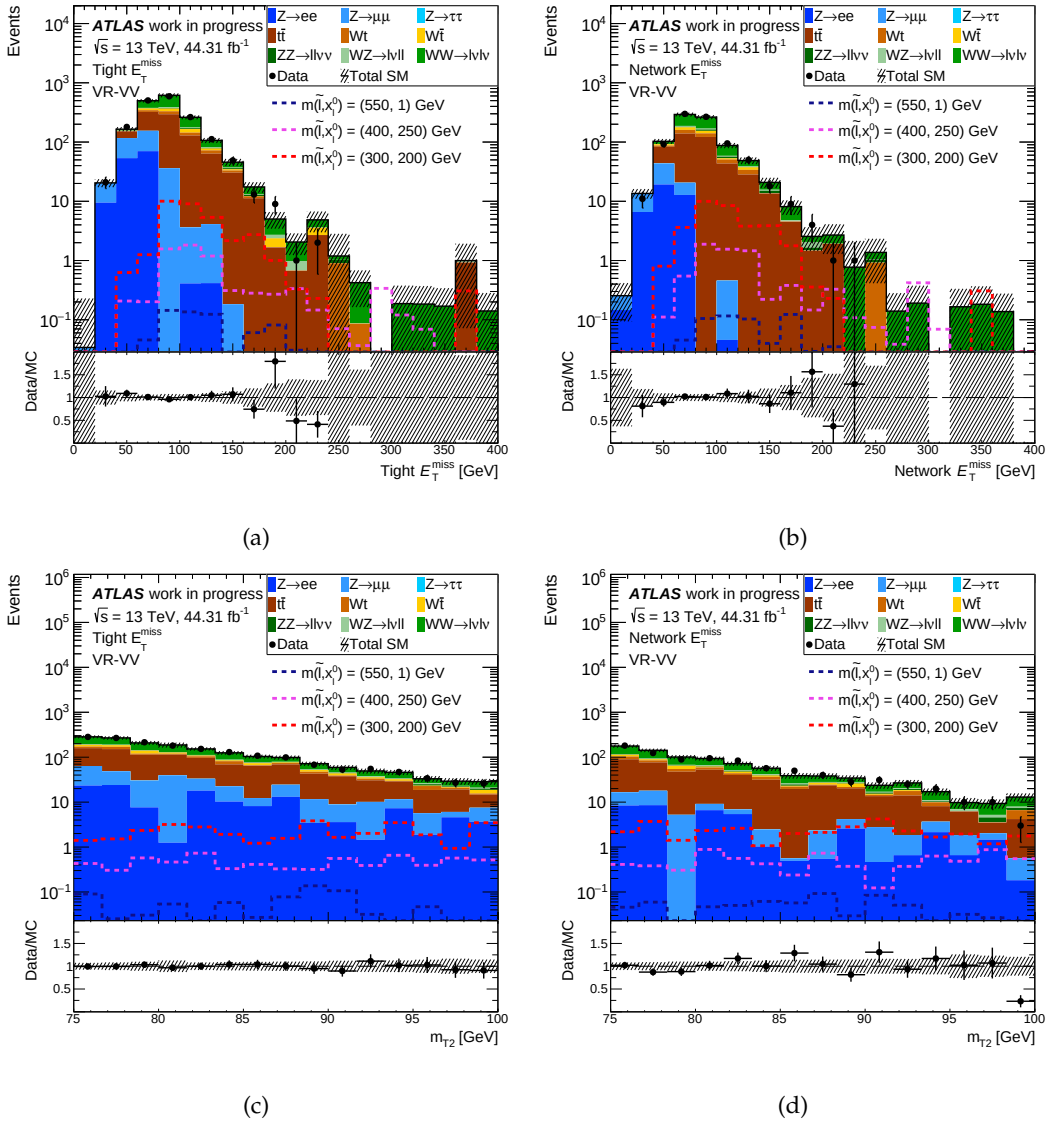
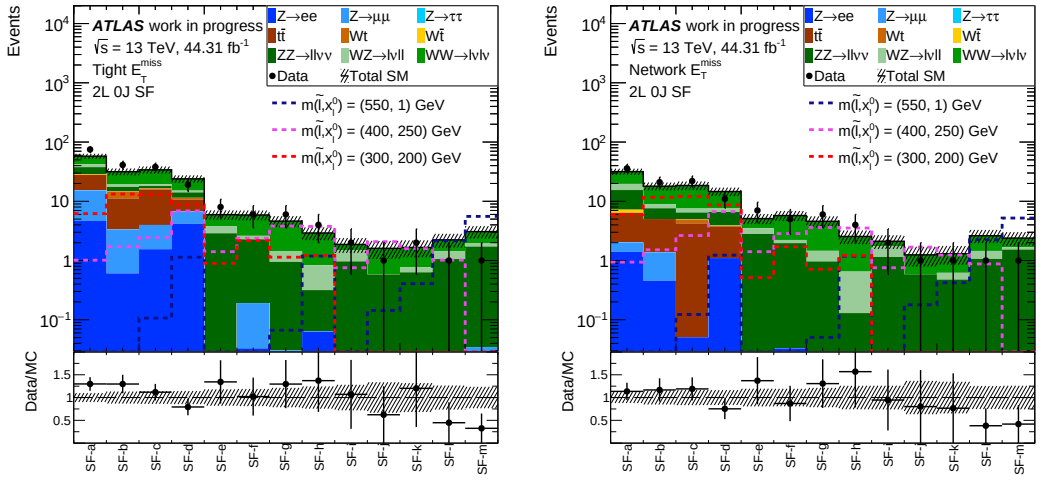


Figure 11.3: Distributions of the reconstructed (a) E_T^{miss} and (c) m_{T2} for data and estimated SM backgrounds in the validation region VR-VV for the Tight study. The same distributions are shown in (b) and (d), respectively, for the Network study. Simulated signal samples are overlaid for comparison.

11.4 Results

The original paper presented a histogram showing the relative distributions of data and SM backgrounds in the 2L-0J-SF signal region using the orthogonal exclusive regions as bins [222]. The equivalent plots for the replicated study using the Tight E_T^{miss} and the Network E_T^{miss} are shown in Figures 11.4(a) and 11.4(b), respectively. Overlaid on these plots are the expected signal events based on the three SUSY samples. Additional plots of the signal region are shown in Figure 11.5. Overall there is good agreement between MC and data for both studies, showing that the neural network did not noticeably cause mismodelling which was a concern raised in Section 10.1.1. This agreement also infers that no noticeable excess was observed and there is no indication that SUSY particles were produced in the data. For both studies,



(a) Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ (b) Network $E_{\mathrm{T}}^{\mathrm{miss}}$

Figure 11.4: The observed and expected SM background yields in the exclusive signal regions for the (a) Tight and the (b) Network study. Simulated signal distributions are overlaid for comparison.

the integrals of data and SM backgrounds agree with one another, as shown in Table 11.3.

Considering the results shown in Table 11.3, the total SM background when using the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ decreased, from 177.12 ± 9.07 to 110.50 ± 6.11. This corresponds to a reduction of 38%. The $Z \rightarrow ll$ events, which were only present in the SR due to large amounts of fake $E_{\mathrm{T}}^{\mathrm{miss}}$, were the most reduced, by 87%. For the processes where m_{T2} has an upper limit lower than the SR requirements, the WW and top backgrounds were diminished by 20% and 59% respectively. The more substantial drop in top events is probably because the increase in resolution achieved by the neural network over the other working points was seen to be the most consequential in events with higher jet multiplicities, as shown in Section 10.2.1. While still noticeable, the other diboson backgrounds did not decrease as meaningfully, with a 10% and 4% reduction in WZ and ZZ events respectively. However, as shown by Figure 11.2, these events can produce genuine m_{T2} values within the range of the SR and are therefore not expected to be reduced by improved $E_{\mathrm{T}}^{\mathrm{miss}}$ performance. The measured $W\bar{t}$ background was the only one that did not decrease in the Network study, but both $W\bar{t}$ contributions in the Tight and Network studies are nearly consistent with zero.

The expected number of signal events as estimated by the three SUSY samples were not considerably reduced in the Network study. Using the SorB as a metric for signal significance $\mathcal{S}_S$, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ led to an overall increase in sensitivity of 26%, 22%, and 16% for the three samples in the inclusive 2L-0J-SF signal region, as shown in Table 11.4. Since the difference between these two studies was in the method of

2L 0J SF	Tight	Network	Relative Change
Observed	204.00 ± 14.85	118.00 ± 11.12	$\approx -42\%$
Total SM	177.12 ± 9.07	110.50 ± 6.11	$\approx -38\%$
$Z \to ee$	11.08 ± 2.90	3.07 ± 0.95	$\approx -72\%$
$Z \to \mu\mu$	18.45 ± 2.99	0.72 ± 1.08	$\approx -96\%$
$t\bar{t}$	36.18 ± 5.65	15.36 ± 3.94	$\approx -58\%$
Wt	5.06 ± 2.02	0.72 ± 0.72	$\approx -86\%$
$W\bar{t}$	0.81 ± 0.81	0.97 ± 0.85	$\approx +20\%$
$ZZ \to llvv$	27.15 ± 2.12	26.04 ± 2.08	$\approx -4\%$
$WZ \to lvll$	12.18 ± 0.99	10.85 ± 0.93	$\approx -11\%$
$WW \to lvlv$	66.28 ± 3.61	52.84 ± 3.16	$\approx -20\%$
$m(\tilde{l}, x_l^0) = (550, 1)$ GeV	10.85 ± 0.53	10.73 ± 0.53	$\approx -1\%$
$m(\tilde{l}, x_l^0) = (400, 250)$ GeV	28.87 ± 1.69	27.74 ± 1.65	$\approx -4\%$
$m(\tilde{l}, x_l^0) = (300, 200)$ GeV	46.50 ± 3.96	42.55 ± 3.78	$\approx -8\%$

Table 11.3: SM background and SUSY signal results in the 2L-0J-SF inclusive region for both the Tight and Network studies.

E_T^{miss} reconstruction and therefore the measured values of m_{T2}, the individual exclusive signal regions were grouped by their m_{T2} cut. The event counts in each group are shown for the Tight study in Table 11.5 and for the Network study in Table 11.6. The Network E_T^{miss} led to an increase in $\mathcal{S}_S$ for each combination of signal sample and grouped signal region, as shown in Table 11.7, with two exceptions. The sensitivity of the $m(\tilde{l}, x_l^0) = (400, 250)$ GeV sample decreased in the bins corresponding to $200\,\text{GeV} < m_{T2} < 300\,\text{GeV}$, and the sensitivity of the $m(\tilde{l}, x_l^0) = (300, 200)$ GeV sample decreased in the bins corresponding to $150\,\text{GeV} < m_{T2} < 200\,\text{GeV}$. However, the upper limits of the genuine m_{T2} for these samples are close to the minimal value of these bins. Therefore, a reduction in event count for these particular bins is entirely possible when using a more accurate E_T^{miss} reconstruction method.

So in conclusion, this chapter showed that the use of the neural network defined E_T^{miss} in a typical analysis led to a significant reduction in backgrounds approximately maintaining the same signal efficiency. Furthermore, the good agreement between data and SM in Figures 11.3-11.5 and Figures A.10-A.12 show that Network E_T^{miss} did not adversely affect the modelling accuracy. Therefore, the use of Network E_T^{miss} has considerable potential to improve the sensitivity of searches for physics beyond the SM.

2L 0J SF	Tight $\mathcal{S}_S$	Network $\mathcal{S}_S$	Increase
$m(\tilde{l}, x_l^0) = (550, 1)$ GeV	0.81 ± 0.05	1.02 ± 0.06	$\approx 26\%$
$m(\tilde{l}, x_l^0) = (400, 250)$ GeV	2.17 ± 0.14	2.64 ± 0.17	$\approx 22\%$
$m(\tilde{l}, x_l^0) = (300, 200)$ GeV	3.49 ± 0.31	4.05 ± 0.38	$\approx 16\%$

Table 11.4: The signal sensitivities, calculated using the signal-over-root-background, for three different SUSY samples in the 2L-0J-SF inclusive region. Values for both the Tight study and the Network study are shown as well as the relative increase.

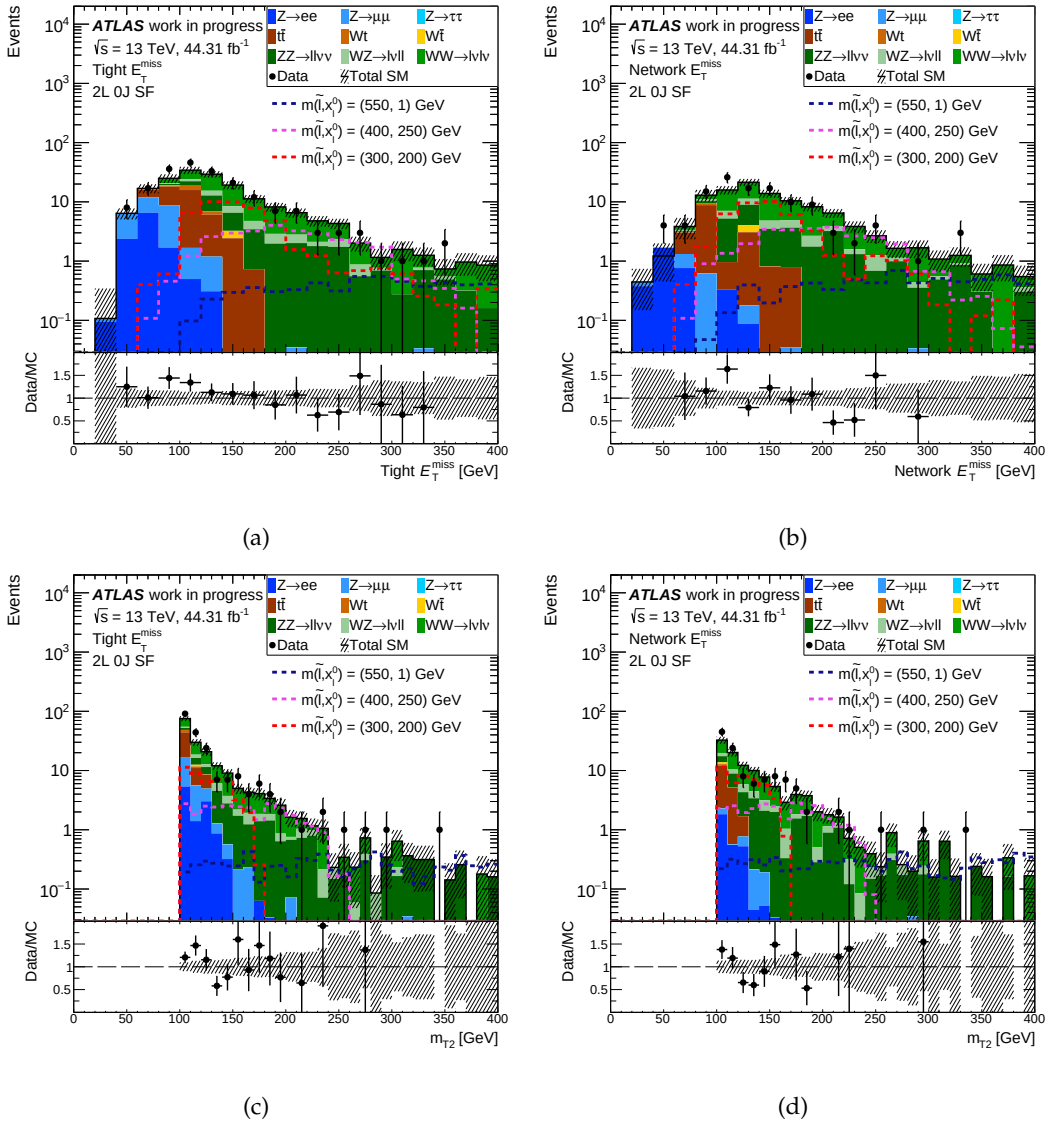


Figure 11.5: Distributions of the reconstructed (a) $E_{\mathrm{T}}^{\mathrm{miss}}$ and (c) $\mathrm{m_{T2}}$ for data and estimated SM backgrounds in the 2L-0J-SF inclusive region, after the application of CR derived normalisation factors for both top and diboson processes. The same distributions are shown in (b) and (d), respectively, for the Network study. Simulated signal distributions are overlaid for comparison.

SR	**SF-(a-d)**	**SF-(e-h)**	**SF-(i-l)**	**SF-m**
Observed	147.29 ± 8.58	19.38 ± 1.81	7.37 ± 1.07	3.08 ± 0.70
Total SM	173.00 ± 13.60	24.00 ± 4.92	6.00 ± 2.45	1.00 ± 1.00
$Z \to ee$	10.98 ± 2.90	0.09 ± 0.08	0	0
$Z \to \mu\mu$	18.27 ± 2.97	0.12 ± 0.33	0.02 ± 0.09	0.03 ± 0.03
$t\bar{t}$	36.18 ± 5.65	0	0	0
Wt	5.06 ± 2.02	0	0	0
$W\bar{t}$	0.81 ± 0.81	0	0	0
$ZZ \to llvv$	15.97 ± 1.60	6.21 ± 0.99	3.28 ± 0.71	1.69 ± 0.51
$WZ \to lvll$	8.65 ± 0.83	2.26 ± 0.42	1.05 ± 0.28	0.22 ± 0.13
$WW \to lvlv$	51.37 ± 3.13	10.69 ± 1.38	3.09 ± 0.72	1.14 ± 0.45
$\mathrm{m}(\tilde{l}, x_l^0) = (550, 1)$ GeV	1.26 ± 0.17	1.29 ± 0.17	2.73 ± 0.24	5.56 ± 0.37
$\mathrm{m}(\tilde{l}, x_l^0) = (400, 250)$ GeV	12.05 ± 1.04	11.38 ± 1.05	5.43 ± 0.66	0
$\mathrm{m}(\tilde{l}, x_l^0) = (300, 200)$ GeV	41.09 ± 3.75	5.42 ± 1.21	0	0

Table 11.5: SM background and signal results in the exclusive signal regions grouped by $\mathrm{m_{T2}}$ cut for the Tight study.

SR	**SF-(a-d)**	**SF-(e-h)**	**SF-(i-l)**	**SF-m**
Observed	90.00 ± 9.66	22.00 ± 4.71	5.00 ± 2.24	1.00 ± 1.00
Total SM	82.81 ± 5.55	18.00 ± 1.75	7.29 ± 1.05	2.39 ± 0.60
$Z \to ee$	3.01 ± 0.95	0.06 ± 0.04	0	0
$Z \to \mu\mu$	1.02 ± 1.04	0.33 ± 0.01	0.03 ± 0.03	0
$t\bar{t}$	15.36 ± 3.94	0	0	0
Wt	0.72 ± 0.72	0	0	0
$W\bar{t}$	0.97 ± 0.85	0	0	0
$ZZ \to llvv$	15.07 ± 1.56	6.14 ± 0.98	3.31 ± 0.72	1.53 ± 0.49
$WZ \to lvll$	7.59 ± 0.77	1.98 ± 0.39	1.12 ± 0.29	0.15 ± 0.11
$WW \to lvlv$	39.09 ± 2.67	10.15 ± 1.34	2.89 ± 0.70	0.71 ± 0.34
$\mathrm{m}(\tilde{l}, x_l^0) = (550, 1)$ GeV	1.37 ± 0.18	1.27 ± 0.17	2.87 ± 0.26	5.21 ± 0.35
$\mathrm{m}(\tilde{l}, x_l^0) = (400, 250)$ GeV	11.78 ± 1.03	11.41 ± 1.06	4.55 ± 0.58	0
$\mathrm{m}(\tilde{l}, x_l^0) = (300, 200)$ GeV	38.39 ± 3.61	4.16 ± 1.06	0	0

Table 11.6: SM background and signal results in the exclusive signal regions grouped by $\mathrm{m_{T2}}$ cut for the Network study.

SR		**SF-(a-d)**	**SF-(e-h)**	**SF-(i-l)**	**SF-m**
$\mathrm{m}(\tilde{l}, x_l^0) = (550, 1)$ GeV	T	0.10 ± 0.01	0.29 ± 0.04	1.01 ± 0.12	3.17 ± 0.42
	N	0.15 ± 0.02	0.30 ± 0.04	1.06 ± 0.12	3.37 ± 0.48
$\mathrm{m}(\tilde{l}, x_l^0) = (400, 250)$ GeV	T	0.99 ± 0.09	2.58 ± 0.27	2.00 ± 0.28	0
	N	1.29 ± 0.12	2.69 ± 0.28	1.69 ± 0.25	0
$\mathrm{m}(\tilde{l}, x_l^0) = (300, 200)$ GeV	T	3.39 ± 0.32	1.23 ± 0.28	0	0
	N	4.22 ± 0.42	0.98 ± 0.25	0	0

Table 11.7: The measured signal sensitivities in the exclusive signal regions grouped by $\mathrm{m_{T2}}$ cut for the Tight (T) and Network (N) studies. The sensitivities are measured using the signal-over-root-background.

Chapter 12

Conclusion

This thesis investigated the use of a deep neural network for $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction in ATLAS. Since it provides a proxy for the measurement of otherwise undetected particles, $E_{\mathrm{T}}^{\mathrm{miss}}$ is one of the most widely used variables in analyses for both SM and BSM physics. However, due to its complexity and the sheer number of signals required to reconstruct the variable, it is extremely sensitive to the misidentification of particles, miscalibration, mismeasurement of particle momenta and the contaminating effects of pileup interactions. The initial aim of the project was to develop a new algorithm, based on a deep neural network, which would produce the most accurate $E_{\mathrm{T}}^{\mathrm{miss}}$ measurements compared to several existing methods at ATLAS, regardless of final state and event topology.

Over the course of this project, almost 3000 different neural networks were trained using MC simulated samples and compared in order to find the optimal network configuration. The effects of some of the most popular techniques in deep learning were studied, such as dropout, batch normalisation, and adaptive learning. 65 different event observables were motivated to be used as inputs for the neural network. The effects of data pre-processing such as standardisation and symmetry removal were also studied. The final model was trained for approximately 100 hours on specialised hardware using 8837525 simulated events of various SM processes, defining a new working point named Network $E_{\mathrm{T}}^{\mathrm{miss}}$.

The performance of the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ was evaluated using the standard techniques and practices used by ATLAS to evaluate any new $E_{\mathrm{T}}^{\mathrm{miss}}$ algorithm. Performance metrics included the $E_{\mathrm{T}}^{\mathrm{miss}}$ resolution, scale and response, angular resolution, tail fraction, and separation power between events with fake $E_{\mathrm{T}}^{\mathrm{miss}}$ and genuine $E_{\mathrm{T}}^{\mathrm{miss}}$. These studies were performed on both $43\,\mathrm{fb}^{-1}$ of pp collision data at $\sqrt{s} = 13\,\mathrm{TeV}$ captured by ATLAS in 2017, as well as a collection of MC simulated samples. Tests were also performed in final states without neutrinos ($Z \to ll$), and in final states that contained neutrinos and varying levels of jet activity ($t\bar{t}$, $WW \to l\nu l\nu$, $H \to WW$).

In every studied topology, in both data and MC, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ produced the best resolution, angular resolution and tail fraction compared to the other algorithms currently used by ATLAS. It also exhibited the greatest separation power, even when

compared to the object-based $E_{\mathrm{T}}^{\mathrm{miss}}$ significance. The Network $E_{\mathrm{T}}^{\mathrm{miss}}$ was also shown to be more resilient to pileup than all other object-based reconstruction methods. Results suggest that the features most modified by the network were the jet and soft-terms.

Having trained on select MC samples, the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ was also observed to transition well to data and other SM processes. It was discovered that a disagreement between $E_{\mathrm{T}}^{\mathrm{miss}}$ in MC and data could arise if the MC already possessed a certain degree of mismodelling in some of the 65 input variables. This was seen when the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ performed better in MC samples of $Z \rightarrow ll$ generated using POWHEG-PYTHIA than in a corresponding selection of events extracted from data. This was determined to be because the generator underestimated the jet multiplicity in the sample.

It was also found that the network produced $E_{\mathrm{T}}^{\mathrm{miss}}$ magnitudes with a loss of response ranging between 9% and 15% depending on the event topology and the value of True $E_{\mathrm{T}}^{\mathrm{miss}}$. This prompted an in-depth investigation on how the distribution shape of $E_{\mathrm{T}}^{\mathrm{miss}}$ in the training set was a primary cause for this negative bias. Attempts to correct this bias by oversampling certain events in the training set could not be done without degrading the overall resolution.

The application of the final model was tested in a search for evidence of SUSY particles based off a previous study conducted by the ATLAS collaboration [222]. The study was performed independently using the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ and the Tight $E_{\mathrm{T}}^{\mathrm{miss}}$ working points. No indication of the production of SUSY particles was observed. However, use of the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ resulted in a considerable reduction of SM backgrounds in the signal region while maintaining similar levels of signal efficiency. This was measured using three simulated samples of $\tilde{l}\tilde{l}$ production. The signal significance of these three samples were much higher in the analysis using the Network $E_{\mathrm{T}}^{\mathrm{miss}}$, corresponding to a relative increase of 26%, 22%, and 16%.

Therefore, the findings in this dissertation show that measurements of $E_{\mathrm{T}}^{\mathrm{miss}}$ predicted by a trained neural network can offer higher reconstruction accuracy, especially as the luminosity of the LHC increases, and can be substantially beneficial for studies of SM processes and for increasing the sensitivity of searches for BSM physics.

12.1 Future Work

There are many possible areas of research which could increase both the accuracy of the neural network and the understanding of how the model arrives at an output.

All jets used in this dissertation were reconstructed from 3D topological clusters of energy deposits in the calorimeter, as is the standard for many analyses at ATLAS. However, a new jet reconstruction technique that follows Particle Flow (PF) [254] has recently been made available. This different approach to jet reconstruction suppresses calorimeter energy deposits from charged pileup particles, and utilises the superior momentum resolution of the ID whenever possible. It better reconstructs the energy flow of the event, and has been shown to improve both jet and $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions [9]. Unfortunately, the software release used in this project did not support the inclusion of PF. Porting the project framework to a newer release, and expanding the list of network inputs to include both PF $E_{\mathrm{T}}^{\mathrm{miss}}$ and PF jet estimates would be a good continuation of this work.

The relationship between the True $E_{\mathrm{T}}^{\mathrm{miss}}$ distribution in the training set and the loss of response exhibited by the Network $E_{\mathrm{T}}^{\mathrm{miss}}$ needs to be further investigated. Mentioned in Section 10.3.2, the use of SUSY samples with high $E_{\mathrm{T}}^{\mathrm{miss}}$ in the training set may lead to better a response without sacrificing resolution. This ties in to another possible addition to this work. One of the major drawbacks experienced during these investigations was the limited number of available datasets. With larger and more diverse training sets, accuracy can only be expected to increase.

The results in Chapter 10 suggest that features of particular importance to the neural network were the $E_{\mathrm{T}}^{\mathrm{miss}}$ jet and soft-terms. This assumption can be tested with a more committed study to determine input significance. However, there is no single measure of predictive importance that is applicable in all situations [255]. One method is to simply retrain the entire network with each input removed, and monitor the resulting change in accuracy. This process would also assist in the investigation of the modelling capabilities of the network when using variables that are themselves difficult to recreate in MC, such as the jet multiplicity which was discussed in Section 10.1.1. However, the training time of the final model exceeded 100 hours, this method was not feasible. The topic gets more complex when the inputs are not independent from one another, as is the case in this project, since the effects of different inputs cannot generally be separated. The most promising method would be to group the inputs into overlapping sets based on their dependencies. For example, several of the 65 inputs to the network depended on the total momentum of the muons. By propagating the gradient of these variables, through the collection of dependent inputs, and through to the output of the network, one can estimate their relative importance. However, input importance would be expected to change depending on event composition.

A noticeable absence in this project was the propagation of systematic uncertainties through the neural network. While one can in principle use systematic variations of the inputs to generate a range of outputs, this does not reflect the true confidence interval of the prediction. The network exploits many correlations which might not be covered by the systematics, so it would be important to compare the data/MC agreement across many topologies.

To fully provide reliable uncertainty intervals on the output of this neural network, like any tool, it needs to be considered as having its own source of uncertainty. If the inputs to the neural network were measured with absolute precision, the output of a trained model should still have some associated error. One of the reasons of this intrinsic uncertainty is due to out-of-distribution data. If a model has been trained exclusively on events with topology-A, and is then tested on events with topology-B, it should be able to convey that these inputs lie outside the collection of familiar examples. Recent advancements in deep learning have shown that models can be developed to intrinsically impart and return uncertainties on their outputs [130]. These are called Bayesian neural networks (BNN). BNNs place a prior distribution over a neural network's weights and biases, and then learn posterior distributions given training data and a process called variational inference [256]. Even the inputs to BNNs can be provided, not as single values, but as probability distributions. Therefore, the network can use the systematic uncertainties of the inputs directly during training to form a better understanding of the reliability of its own estimates. However, developing and training a BNN would require rebuilding most of the framework used in this project, and it is thus left as potential future work.

Finally, machine learning is one of the fastest moving fields in data science today. Papers introducing new techniques are regularly released that drastically change the landscape of research and set new standards for training practices. Examples of these include the introduction of dropout [129], convolutional layers [15], adaptive learning methods like Adam [127], and batch normalisation [139]. By the time that this dissertation is printed, many new and exciting developments would have been made. These have the potential to further improve the already impressive gains found in this project of applying deep learning techniques to $E_{\mathrm{T}}^{\mathrm{miss}}$ reconstruction.

Appendix A

Additional Plots and Figures

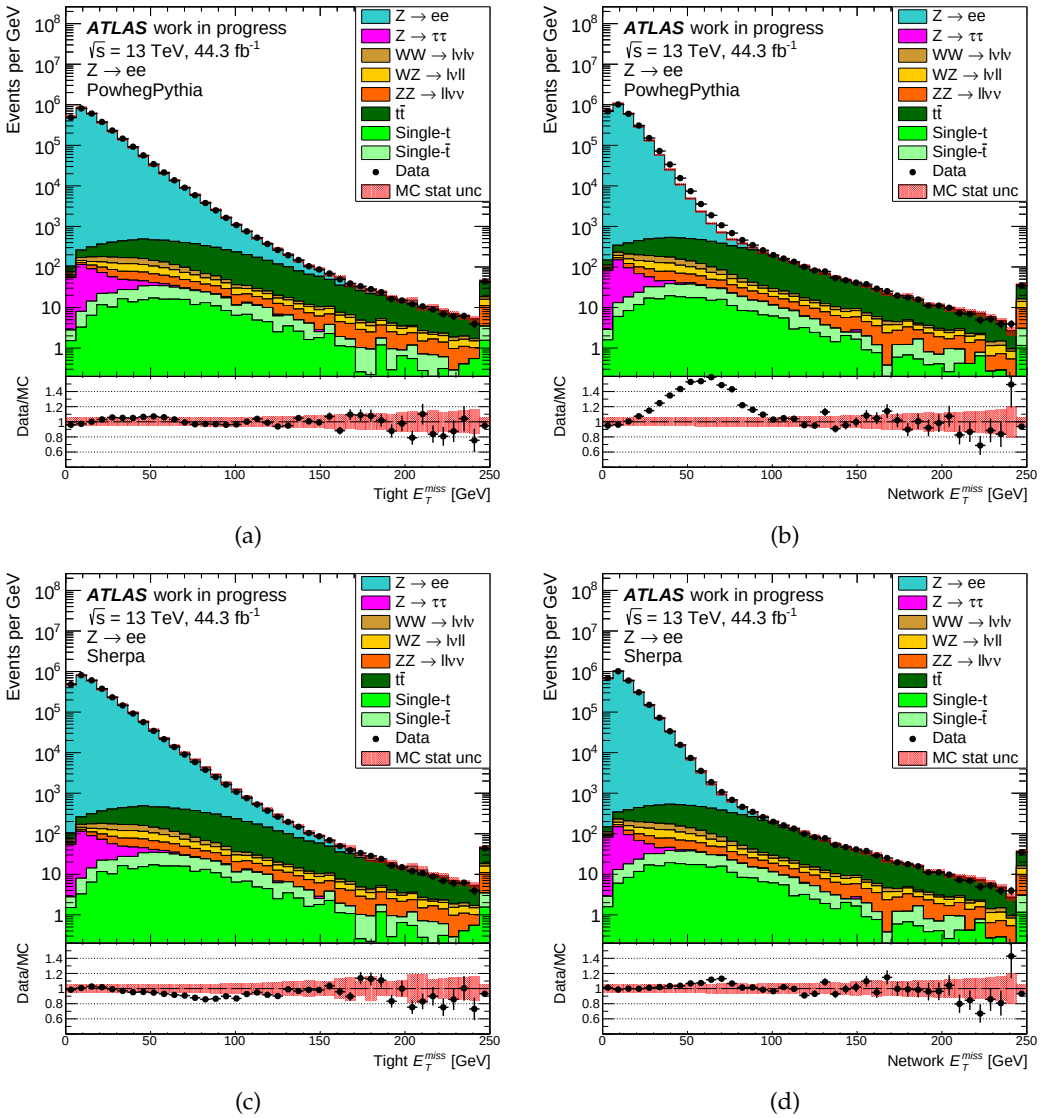


Figure A.1: Distributions of $E_{\mathrm{T}}^{\mathrm{miss}}$ using the (a) Tight and (b) Network working points for an inclusive sample of $Z \rightarrow ee$ events where the MC signal sample was generated using POWHEG. The same distributions using SHERPA for the signal sample are shown in (c) and (d), respectively. The shaded areas indicate the uncertainty for MC simulations without systematic contributions. The last bin of each plot includes the overflow.

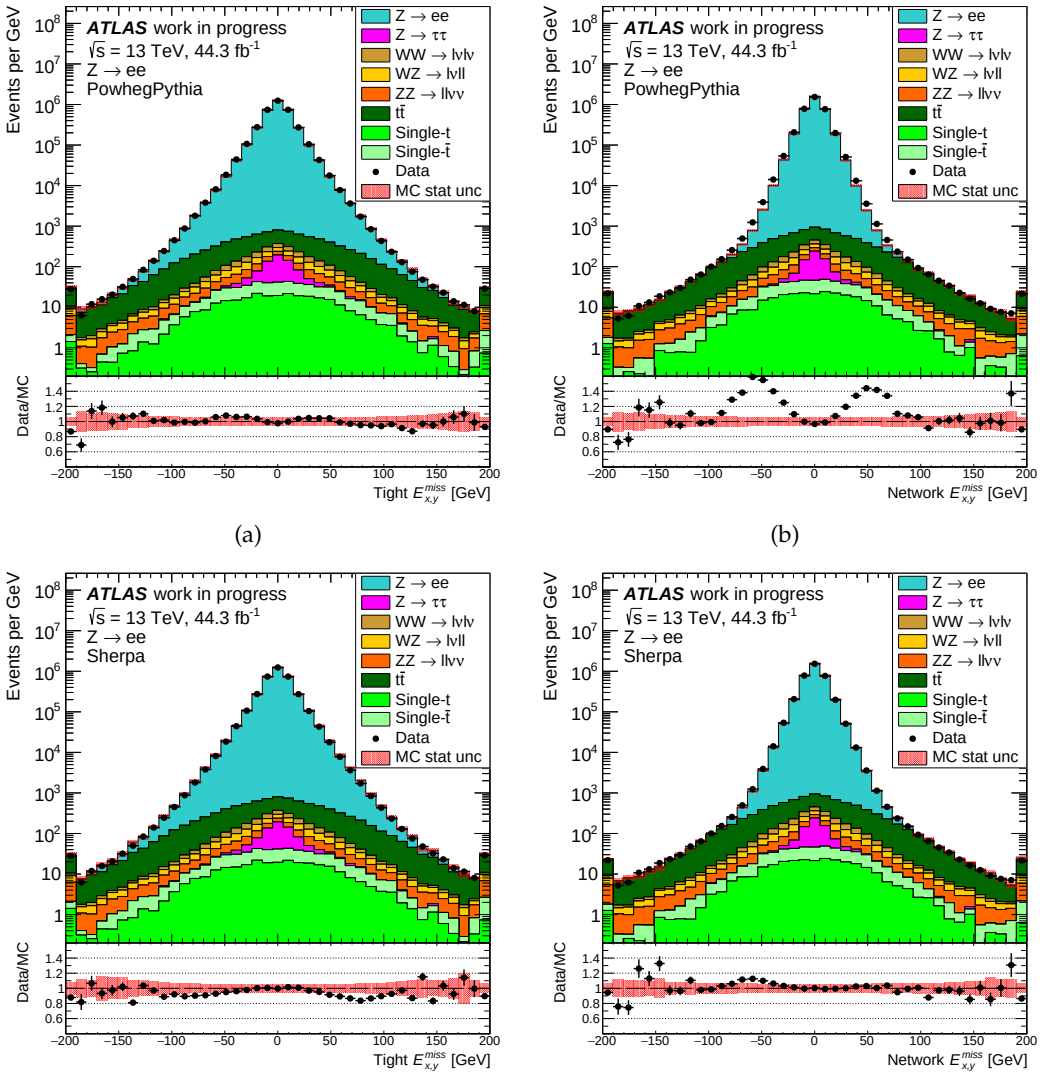
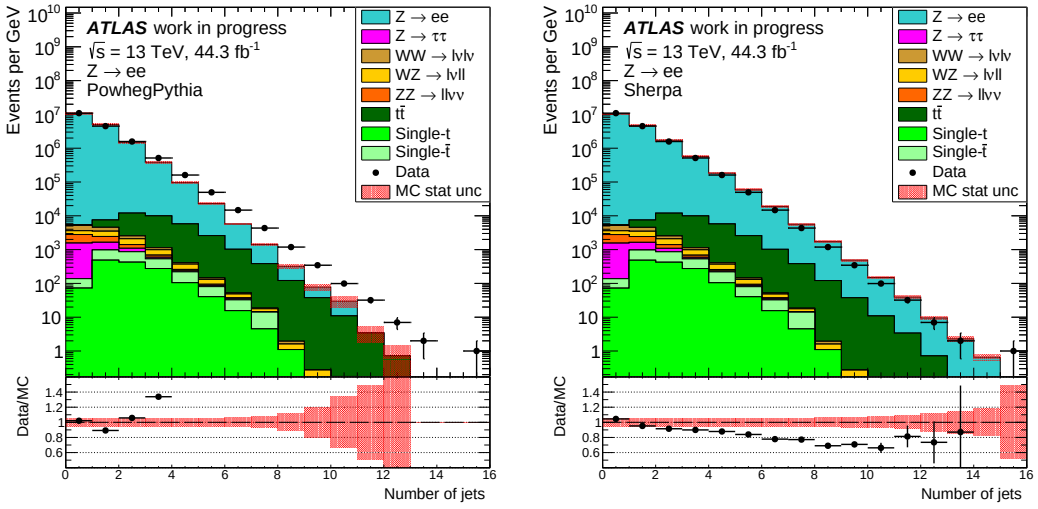
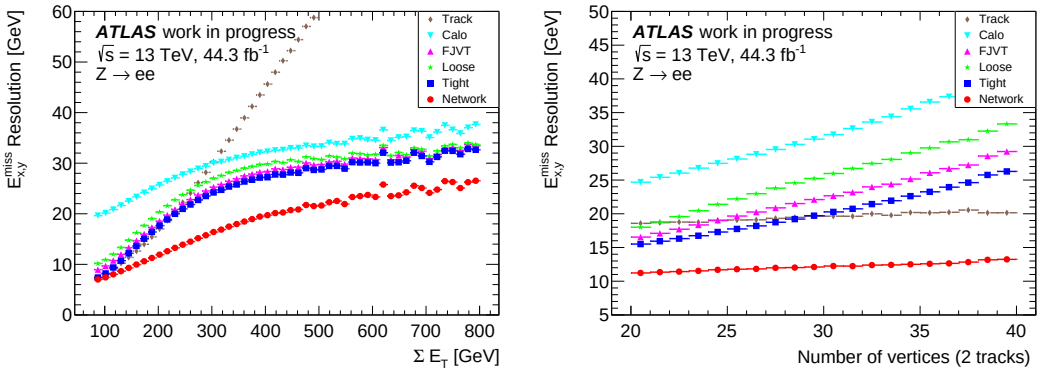


Figure A.2: Distributions of $E_{x,y}^{\text{miss}}$ using the (a) Tight and (b) Network working points for an inclusive sample of $Z \rightarrow ee$ events where the MC signal sample was generated using POWHEG. The same distributions using SHERPA for the signal sample are shown in (c) and (c), respectively. The shaded areas indicate the uncertainty for MC simulations without systematic contributions. The first and last bin of each plot includes the underflow and overflow, respectively.



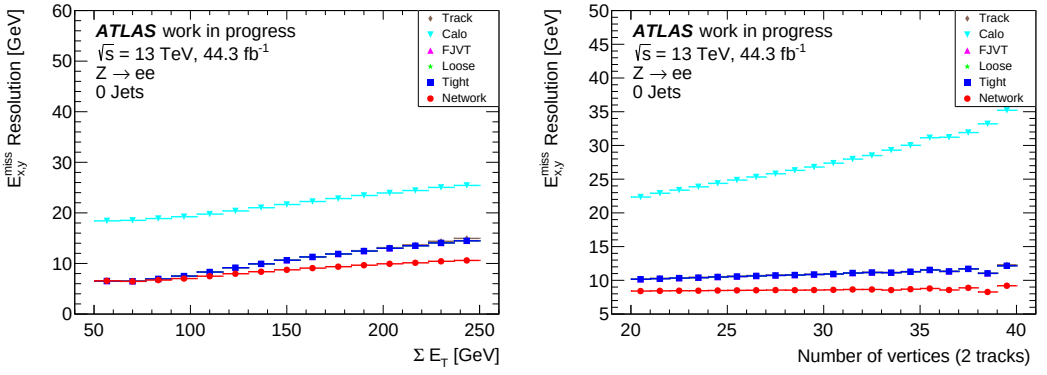
(a) (b)

Figure A.3: Distributions of the jet multiplicity for an inclusive sample of $Z \rightarrow ee$ events where the signal events were generated using (a) POWHEG and (b) SHERPA. The shaded areas indicate the uncertainty for MC simulations without systematic contributions.



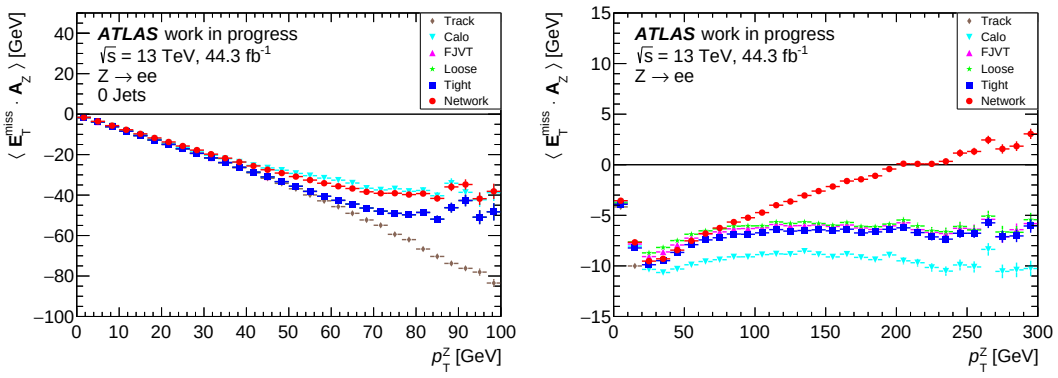
(a) (b)

Figure A.4: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions of six working points in (a) bins of Tight ΣE_{T} and (b) bins of the number of reconstructed primary vertices on an inclusive sample of $Z \rightarrow ee$ events extracted from data.



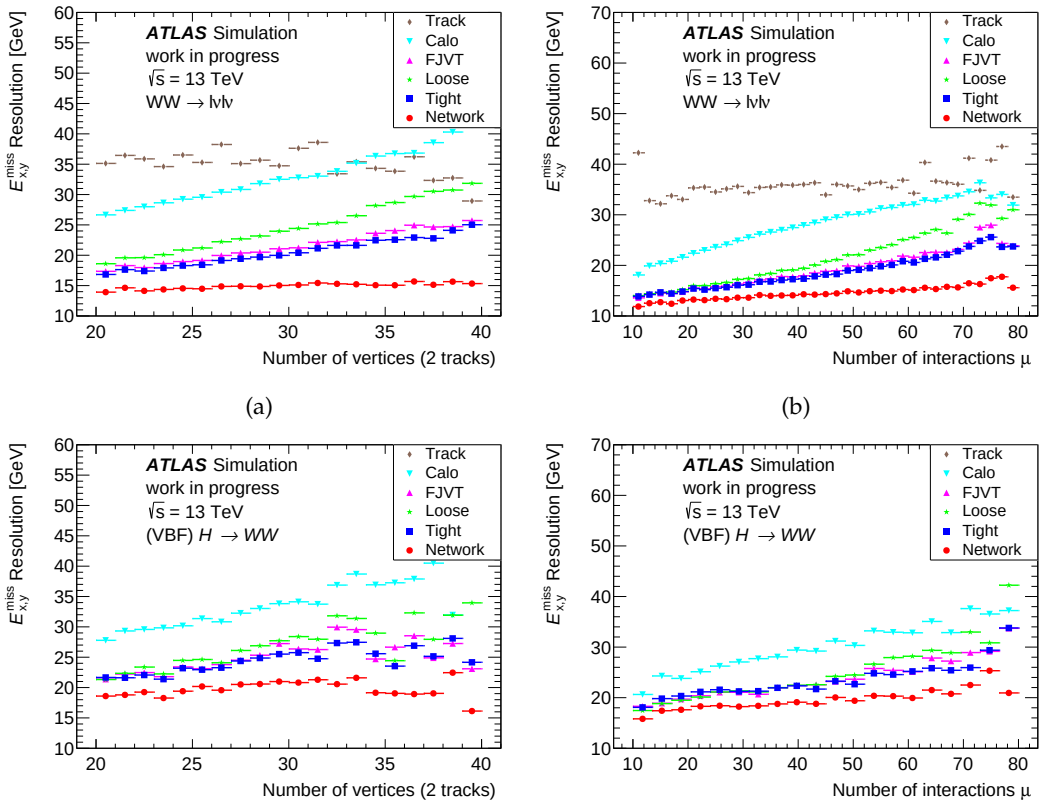
(a) (b)

Figure A.5: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions of six working points in (a) bins of Tight ΣE_{T} and (b) bins of the number of reconstructed primary vertices on a 0-jet sample of $Z \rightarrow ee$ events extracted from data.



(a) (b)

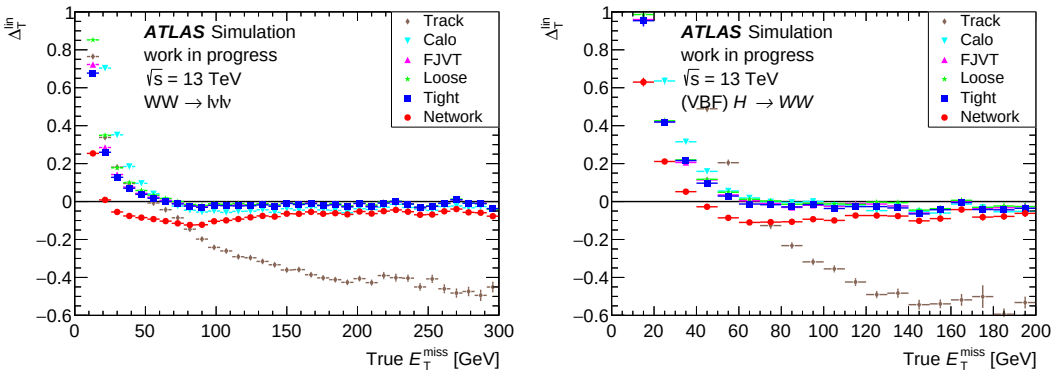
Figure A.6: Plots showing $\langle \mathcal{P}_{||}^{Z} \rangle = \langle \mathbf{E}_{\mathrm{T}}^{\mathrm{miss}} \cdot \mathbf{A}_{Z} \rangle$ as a function of p_{T}^{Z} for the (a) 0-Jet and (b) inclusive events in $Z \to ee$ data.



(a) (b)

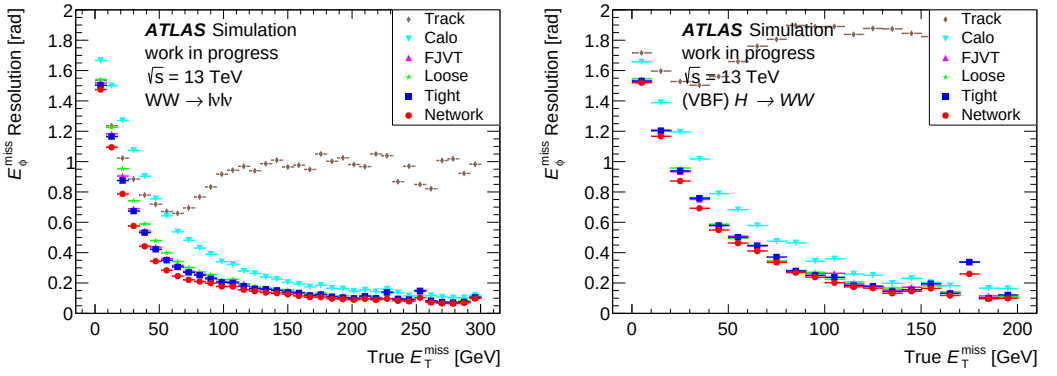
(c) (d)

Figure A.7: The $E_{\mathrm{T}}^{\mathrm{miss}}$ resolutions measured by $\mathrm{RMSE}^{\mathrm{miss}}$ using six different working points in a MC $WW \to lvlv$ sample are shown versus pileup measured by (a) N_{PV} and (b) mu. The same distributions in a (VBF) Higgs sample are shown in (c) and (d) respectively.



(a) (b)

Figure A.8: The deviation of the $E_{\mathrm{T}}^{\mathrm{miss}}$ response from linearity using six different working points measured as a function of the True $E_{\mathrm{T}}^{\mathrm{miss}}$ in (a) $WW \rightarrow l\nu l\nu$ and (a) (VBF) Higgs MC simulations.



(a) (b)

Figure A.9: The angular resolution measured by the RMSE of the reconstructed ϕ^{miss} distribution plotted in bins of True $E_{\mathrm{T}}^{\mathrm{miss}}$ for six working points in a simulated (a) $WW \rightarrow l\nu l\nu$ and (b) (VBF) Higgs sample.

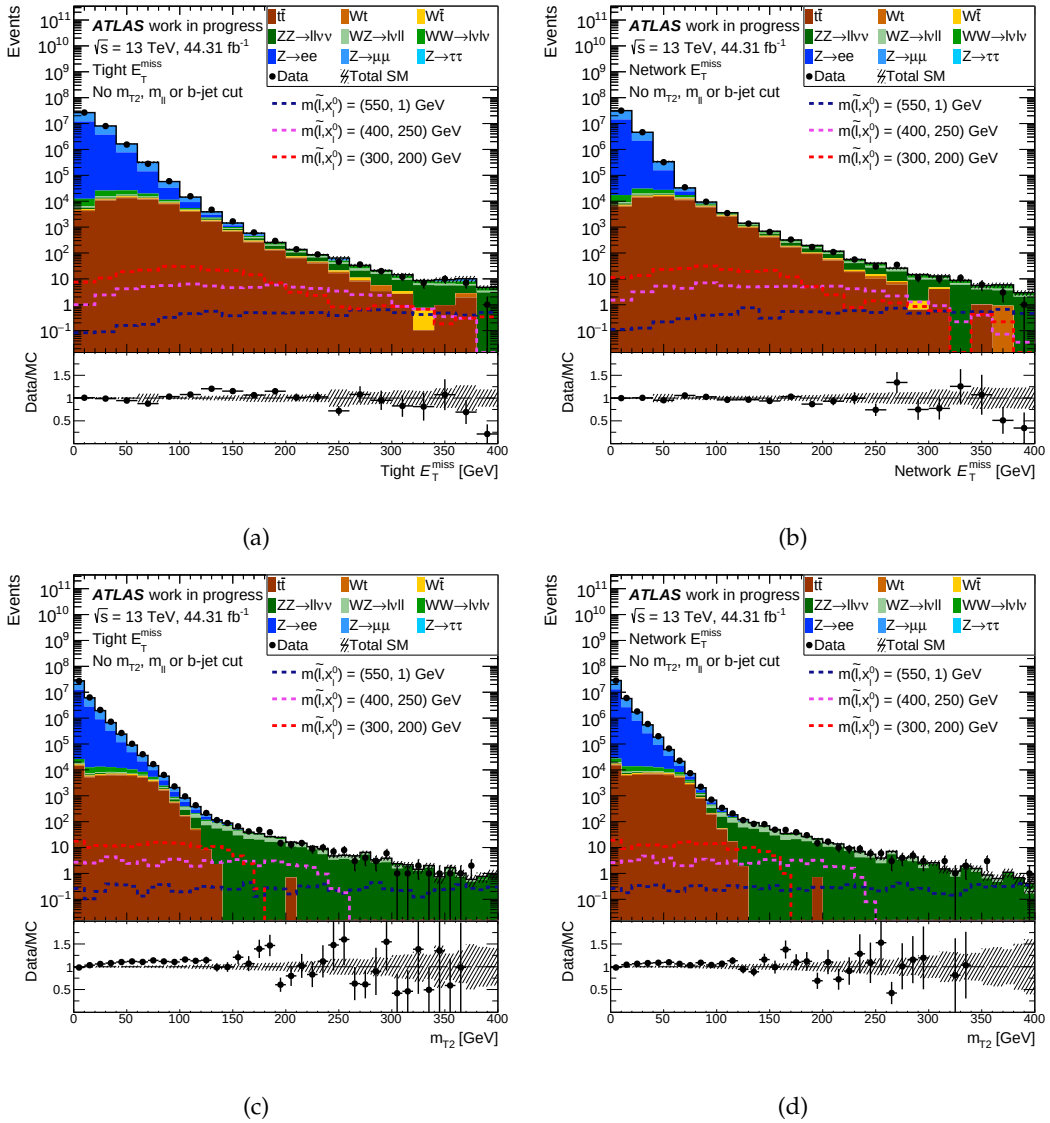
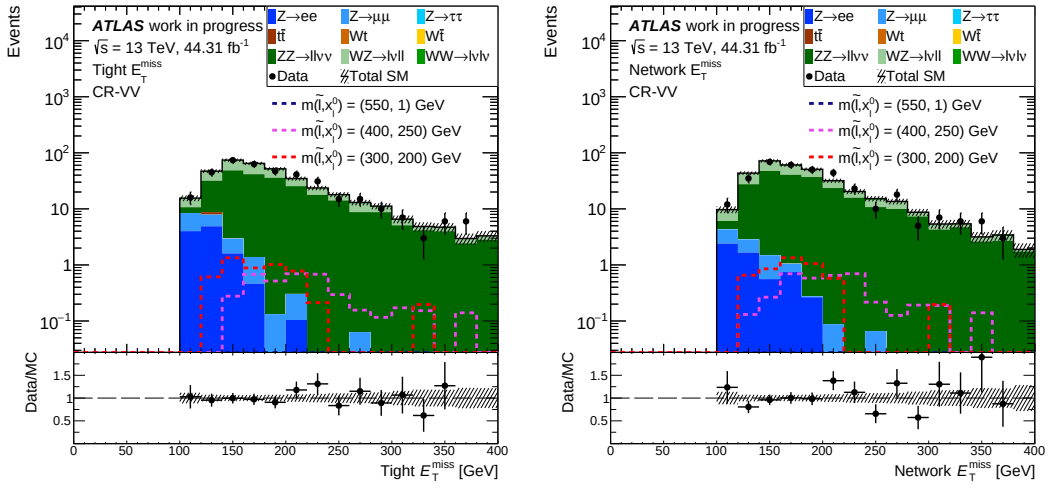


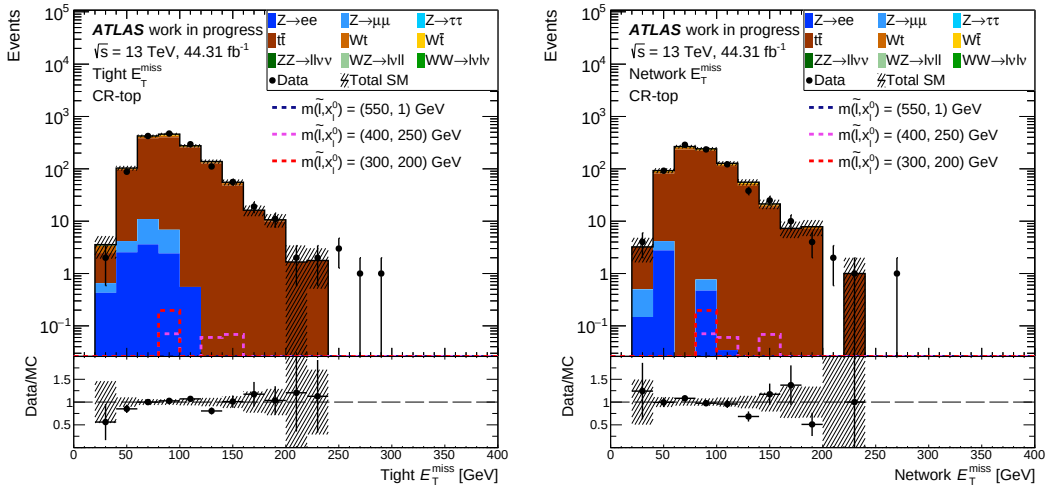
Figure A.10: Distributions of the reconstructed (a) $E_{\mathrm{T}}^{\mathrm{miss}}$ and (c) m_{T2} for data and estimated SM backgrounds after the preselections for 2L-0J-SF. The same distributions are shown in (b) and (d), respectively, for the Network study. Simulated signal samples are overlaid for comparison.



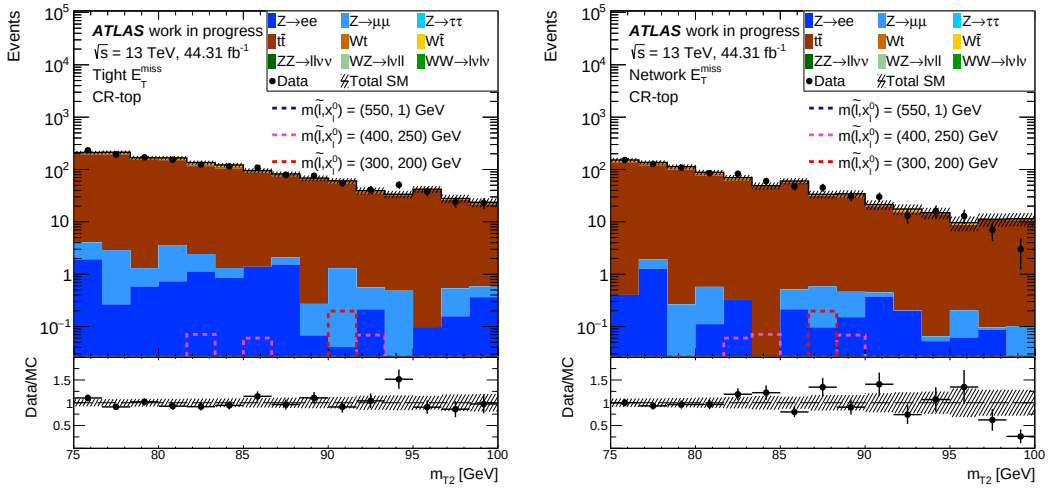
(a) (b)

(c) (d)

Figure A.11: Distributions of the reconstructed (a) $E_{\mathrm{T}}^{\mathrm{miss}}$ and (c) m_{T2} for data and estimated SM backgrounds in CR-VV, after the application of CR derived normalisation factors for both top and diboson processes. The same distributions are shown in (b) and (d), respectively, for the Network study. Simulated signal distributions are overlaid for comparison.



(a) (b)



(c) (d)

Figure A.12: Distributions of the reconstructed (a) $E_{\mathrm{T}}^{\mathrm{miss}}$ and (c) m_{T2} for data and estimated SM backgrounds in CR-top, after the application of CR derived normalisation factors for both top and diboson processes. The same distributions are shown in (b) and (d), respectively, for the Network study. Simulated signal distributions are overlaid for comparison.

References

[1] S. L. Glashow, "Partial Symmetries of Weak Interactions", *Nuclear Physics*, vol. 22, pp. 579–588, 1961.
DOI: 10.1016/0029-5582(61)90469-2.

[2] S. Weinberg, "A Model of Leptons", *Physical Review Letters*, vol. 19, pp. 1264–1266, 1967.
DOI: 10.1103/PhysRevLett.19.1264.

[3] A. Salam and J. Ward, "Electromagnetic and weak interactions", *Physics Letters*, vol. 13, no. 2, pp. 168–171, 1964, ISSN: 0031-9163.
DOI: https://doi.org/10.1016/0031-9163(64)90711-5.

[4] ATLAS Collaboration, "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC", *Physics Letters*, vol. B716, no. 1, pp. 1–29, 2012, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/j.physletb.2012.08.020.

[5] ATLAS Collaboration, "Measurement of the W boson mass in pp collisions at $\sqrt{s} = 7$ TeV with the ATLAS detector", *The European Physical Journal*, vol. C78, no. 2, Feb. 2018, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-017-5475-4.

[6] ATLAS Collaboration, "Measurement of the W^+W^- production cross section in pp collisions at a centre-of-mass energy of $\sqrt{s}$ = 13 TeV with the ATLAS experiment", *Physics Letters*, vol. B773, pp. 354–374, 2017.
DOI: 10.1016/j.physletb.2017.08.047.

[7] B. Pearson, "Top quark mass in ATLAS", in *Proceedings, International Workshop on Top Quark Physics, Braga, Portugal*, Sep. 2017.
arXiv: 1711.09763 [hep-ex].

[8] G. Apollinari *et al.*, *High-Luminosity Large Hadron Collider (HL-LHC): Technical Design Report V. 0.1*, ser. CERN Yellow Reports: Monographs. Geneva: CERN, 2017.
DOI: 10.23731/CYRM-2017-004.

[9] ATLAS Collaboration, "$E_{\mathrm{T}}^{\mathrm{miss}}$ performance in the ATLAS detector using 2015-2016 LHC p-p collisions", CERN, Geneva, Tech. Rep. ATLAS-CONF-2018-023, Jun. 2018.
[Online]. Available: https://cds.cern.ch/record/2625233.

[10] ATLAS Collaboration, "Performance of missing transverse momentum reconstruction with the ATLAS detector using proton–proton collisions at $\sqrt{s} = 13$ TeV", *The European Physical Journal*, vol. C78, no. 11, Nov. 2018, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-018-6288-9.

[11] ATLAS Collaboration, "Performance of algorithms that reconstruct missing transverse momentum in $\sqrt{s} = 8$ TeV proton-proton collisions in the ATLAS detector", *The European Physical Journal*, vol. C77, no. 4, p. 241, 2017.
DOI: 10.1140/epjc/s10052-017-4780-2.

[12] S. Lawrence, C. L. Giles, *et al.*, "Face recognition: A convolutional neural-network approach", *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.
DOI: 10.1109/72.554195.

[13] H. Lu, R. Setiono, and H. Liu, "Effective data mining using neural networks", *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 6, pp. 957–961, 1996.
DOI: 10.1109/69.553163.

[14] I. M. Nasser and S. S. Abu-Naser, "Lung Cancer Detection Using Artificial Neural Network", *International Journal of Engineering and Information Systems*, vol. 3, no. 3, pp. 17–23, Mar. 2019.
[Online]. Available: https://ssrn.com/abstract=3369062.

[15] Y. LeCun, B. Boser, J. S. Denker, *et al.*, "Backpropagation Applied to Handwritten Zip Code Recognition", *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec. 1989, ISSN: 0899-7667.
DOI: 10.1162/neco.1989.1.4.541.

[16] K. Albertsson, P. Altoe, D. Anderson, *et al.*, "Machine Learning in High Energy Physics Community White Paper", *Journal of Physics: Conference Series*, vol. 1085, p. 022 008, Sep. 2018.
DOI: 10.1088/1742-6596/1085/2/022008.

[17] M. Thomson, *Modern Particle Physics*. Cambridge University Press, 2013, ISBN: 9781107034266.

[18] D. Griffiths, *Introduction to elementary particles*. Wiley, 1987, ISBN: 9780471603863.

[19] G. Barr, R. Devenish, *et al.*, *Particle physics in the LHC era*. Oxford: Oxford University Press, 2016, vol. 24, ISBN: 3257227892.
DOI: 10.1093/acprof:oso/9780198748557.001.0001.

[20] N. M. Köhler, "Searches for the Supersymmetric Partner of the Top Quark, Dark Matter and Dark Energy at the ATLAS Experiment", Dissertation, Technical University of Munich, Munich, Germany, 2018.
DOI: 10.1007/978-3-030-25988-4.

[21] Y. Fukuda, T. Hayakawa, E. Ichihara, *et al.*, "Evidence for Oscillation of Atmospheric Neutrinos", *Physical Review Letters*, vol. 81, no. 8, pp. 1562–1567, Aug. 1998, ISSN: 1079-7114.
DOI: 10.1103/physrevlett.81.1562.

[22] Wikipedia contributors, *Standard Model*, General Photo, 2019.
[Online]. Available: https://en.wikipedia.org/wiki/Standard_Model (visited 23/1/2019).

[23] G. 't Hooft and M. J. G. Veltman, "Regularization and Renormalization of Gauge Fields", *Nuclear Physics*, vol. B44, pp. 189–213, 1972.
DOI: 10.1016/0550-3213(72)90279-9.

[24] C.-N. Yang and R. L. Mills, "Conservation of Isotopic Spin and Isotopic Gauge Invariance", *Physical Review Letters*, vol. 96, pp. 191–195, 1954.
DOI: 10.1103/PhysRev.96.191.

[25] E. Noether, "Invariant variation problems", *Transport Theory and Statistical Physics*, vol. 1, no. 3, pp. 186–207, 1971.
DOI: 10.1080/00411457108231446.

[26] W. Tung, *Group Theory in Physics*. World Scientific, 1985, p. 231, ISBN: 9789971966560.
[Online]. Available: https://books.google.co.za/books?id=089tgpOBO04C.

[27] D. J. Gross and F. Wilczek, "Ultraviolet behavior of non-abelian gauge theories", *Physical Review Letters*, vol. 30, no. 26, pp. 1343–1346, 1973.
DOI: 10.1103/PhysRevLett.30.1343.

[28] H. D. Politzer, "Reliable perturbative results for strong interactions", *Physical Review Letters*, vol. 30, no. 26, pp. 1346–1349, 1973.
DOI: 10.1103/PhysRevLett.30.1346.

[29] J. Greensite, *An Introduction to the Confinement Problem*, ser. Lecture Notes in Physics. Springer Berlin Heidelberg, 2011, ISBN: 9783642143816.
[Online]. Available: https://books.google.co.za/books?id=CP7%5C_QooHo8wC.

[30] H. D. Politzer, "Asymptotic Freedom: An Approach to Strong Interactions", *Physics Reports*, vol. 14, pp. 129–180, 1974.
DOI: 10.1016/0370-1573(74)90014-3.

[31] R. P. Feynman, "Mathematical formulation of the quantum theory of electromagnetic interaction", *Physical Review*, vol. 80, no. 3, pp. 440–457, 1950.
DOI: 10.1103/PhysRev.80.440.

[32] C. Wu, E. Ambler, *et al.*, "Experimental test of parity conservation in beta decay", *Physical review*, vol. 105, no. 4, pp. 1413–1414, 1957.
DOI: 10.1103/PhysRev.105.1413.

[33] F. Englert and R. Brout, "Broken symmetry and the mass of gauge vector mesons", *Physical Review Letters*, vol. 13, no. 9, pp. 321–323, 1964.
DOI: 10.1103/PhysRevLett.13.321.

[34] P. W. Higgs, "Broken symmetries and the masses of gauge bosons", *Physical Review Letters*, vol. 13, no. 16, pp. 508–509, 1964.
DOI: 10.1103/PhysRevLett.13.508.

[35] G. S. Guralnik, C. R. Hagen, and T. W. Kibble, "Global conservation laws and massless particles", *Physical Review Letters*, vol. 13, no. 20, pp. 585–587, 1964.
DOI: 10.1103/PhysRevLett.13.585.

[36] CMS Collaboration, "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC", *Physics Letters*, vol. B716, no. 1, pp. 30–61, 2012, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/j.physletb.2012.08.021.

[37] ATLAS Collaboration, "Evidence for the spin-0 nature of the Higgs boson using ATLAS data", *Physics Letters*, vol. B726, no. 1, pp. 120–144, 2013, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/j.physletb.2013.08.026.

[38] CMS Collaboration, "Study of the Mass and Spin-Parity of the Higgs Boson Candidate via Its Decays to Z Boson Pairs", *Physical Review Letters*, vol. 110, p. 081 803, 8 Feb. 2013.
DOI: 10.1103/PhysRevLett.110.081803.

[39] J. Kretzschmar, "Standard Model physics at the LHC", in *From My Vast Repertoire: Guido Altarelli's Legacy*, A. Levy, S. Forte, and G. Ridolfi, Eds., 2019, pp. 153–171.
DOI: 10.1142/9789813238053_0009.

[40] M. Dine and A. Kusenko, "Origin of the matter - antimatter asymmetry", *Review of Modern Physics*, vol. 76, pp. 1–30, 1 Dec. 2003.
DOI: 10.1103/RevModPhys.76.1.

[41] A. D. Sakharov, "Violation of CP invariance, C asymmetry, and baryon asymmetry of the universe", *JETP lett.*, vol. 5, pp. 32–35, 1967.
DOI: 10.1070/PU1991v034n05ABEH002497.

[42] J. Cline, "Status of electroweak phase transition and baryogenesis", *Pramana*, vol. 55, no. 1-2, pp. 33–42, Jul. 2000, ISSN: 0973-7111.
DOI: 10.1007/s12043-000-0081-6.

[43] C. A. Baker, D. D. Doyle, *et al.*, "Improved experimental limit on the electric dipole moment of the neutron", *Physical Review Letters*, vol. 97, no. 13, Sep. 2006, ISSN: 1079-7114.
DOI: 10.1103/physrevlett.97.131801.

[44] J. E. Kim and G. Carosi, "Axions and the strong CP problem", *Reviews of Modern Physics*, vol. 82, pp. 557–602, 2010.
DOI: 10.1103/RevModPhys.91.049902.

[45] M. Pospelov and A. Ritz, "Electric dipole moments as probes of new physics", *Annals of Physics*, vol. 318, no. 1, pp. 119–169, Jun. 2005, ISSN: 0003-4916.
DOI: 10.1016/j.aop.2005.04.002.

[46] G. Burdman, "New solutions to the hierarchy problem", *Brazilian journal of physics*, vol. 37, no. 2B, pp. 506–513, 2007.
DOI: 10.1590/s0103-97332007000400006.

[47] G. 't Hooft, C. Itzykson, *et al.*, *Recent Developments in Gauge Theories. Proceedings, Nato Advanced Study Institute, Cargese, France, August 26 - September 8, 1979*. 1980, vol. B59, pp. 1–438.
DOI: 10.1007/978-1-4684-7571-5.

[48] J. L. Feng, "Dark matter candidates from particle physics and methods of detection", *Annual Review of Astronomy and Astrophysics*, vol. 48, pp. 495–545, 2010.
DOI: 10.1146/annurev-astro-082708-101659.

[49] K. C. Freeman, "On the disks of spiral and S0 galaxies", *The Astrophysical Journal*, vol. 160, p. 811, 1970.
DOI: 10.1086/150474.

[50] V. C. Rubin and W. K. Ford Jr., "Rotation of the Andromeda Nebula from a Spectroscopic Survey of Emission Regions", *Astrophys. J.*, vol. 159, pp. 379–403, 1970.
DOI: 10.1086/150317.

[51] D. Scott, M. J. White, *et al.*, "Cosmological difficulties with modified Newtonian dynamics", *Monthly Notices of the Royal Astronomical Society*, 2001.
arXiv: astro-ph/0104435 [astro-ph].

[52] G. Bertone, D. Hooper, and J. Silk, "Particle dark matter: evidence, candidates and constraints", *Physics Reports*, vol. 405, no. 5, pp. 279–390, 2005, ISSN: 0370-1573.
DOI: https://doi.org/10.1016/j.physrep.2004.08.031.

[53] J. Yoo, J. Chaname, and A. Gould, "The end of the MACHO era: limits on halo dark matter from stellar halo wide binaries", *The Astrophysical Journal*, vol. 601, no. 1, pp. 311–318, 2004.
DOI: 10.1086/380562.

[54] D. Raine and E. Thomas, *An Introduction to the Science of Cosmology*, ser. Series in Astronomy and Astrophysics. CRC Press, 2018, ISBN: 9781351991087.
[Online]. Available: https://books.google.co.za/books?id=GH90DwAAQBAJ.

[55] Y. A. Golfand and E. P. Likhtman, "Extension of the Algebra of Poincare Group Generators and Violation of p Invariance", *JETP Letters*, vol. 13, pp. 323–326, 1971.
[Online]. Available: http://cds.cern.ch/record/433516.

[56] D. Volkov and V. Akulov, "Is the neutrino a goldstone particle?", *Physics Letters*, vol. B46, no. 1, pp. 109–110, 1973, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(73)90490-5.

[57] S. Martin, "A Supersymmetry Primer", in. World Scientific, Jul. 1998, pp. 1–98.
DOI: 10.1142/9789812839657_0001.

[58] J. Wess and B. Zumino, "Supergauge Transformations in Four-Dimensions", *Nuclear Physics*, vol. B70, pp. 39–50, 1974.
DOI: 10.1016/0550-3213(74)90355-1.

[59] J. Wess and B. Zumino, "Supergauge invariant extension of quantum electrodynamics", *Nuclear Physics*, vol. B78, no. 1, pp. 1–13, 1974.
DOI: 10.1016/0550-3213(74)90112-6.

[60] J. Wess and B. Zumino, "Supergauge invariant yang-mills theories", *Nuclear Physics*, vol. B79, no. 3, pp. 413–421, 1974.
DOI: 10.1016/0550-3213(74)90559-8.

[61] A. Salam and J. Strathdee, "Super-symmetry and Nonabelian gauges", *Physics Letters*, vol. B51, no. 4, pp. 353–355, 1974.
DOI: 10.1016/0370-2693(74)90226-3.

[62] P. Fayet, "Supersymmetry and weak, electromagnetic and strong interactions", *Physics Letters*, vol. B64, no. 2, pp. 159–162, 1976, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(76)90319-1.

[63] P. Fayet, "Spontaneously broken supersymmetric theories of weak, electromagnetic and strong interactions", *Physics Letters*, vol. B69, no. 4, pp. 489–494, 1977, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(77)90852-8.

[64] A. Djouadi *et al.*, "The Minimal supersymmetric standard model: Group summary report", in *Proceedings, Groupement De Recherche - Supersymetrie, Montpellier, France*, Apr. 1998.
arXiv: hep-ph/9901246 [hep-ph].

[65] G. R. Farrar and P. Fayet, "Phenomenology of the production, decay, and detection of new hadronic states associated with supersymmetry", *Physics Letters*, vol. B76, no. 5, pp. 575–579, 1978, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(78)90858-4.

[66] H. Goldberg, "Constraint on the Photino Mass from Cosmology", *Physical Review Letters*, vol. 50, pp. 1419–1422, 19 May 1983.
DOI: 10.1103/PhysRevLett.50.1419.

[67] J. Ellis, J. Hagelin, *et al.*, "Supersymmetric relics from the big bang", *Nuclear Physics*, vol. B238, no. 2, pp. 453–476, 1984, ISSN: 0550-3213.
DOI: https://doi.org/10.1016/0550-3213(84)90461-9.

[68] N. Sakai, "Naturalnes in supersymmetric GUTS", *Zeitschrift fur Physik C Particles and Fields*, vol. 11, no. 2, pp. 153–157, Jun. 1981, ISSN: 1431-5858.
DOI: 10.1007/BF01573998.

[69] L. Ibáñez and G. Ross, "Low-energy predictions in supersymmetric grand unified theories", *Physics Letters*, vol. B105, no. 6, pp. 439–442, 1981, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(81)91200-4.

[70] S. Dimopoulos, S. Raby, and F. Wilczek, "Supersymmetry and the scale of unification", *Physical Review*, vol. D24, pp. 1681–1683, 6 Sep. 1981.
DOI: 10.1103/PhysRevD.24.1681.

[71] W. de Boer, "Grand unified theories and supersymmetry in particle physics and cosmology", *Progress in Particle and Nuclear Physics*, vol. 33, pp. 201–302, 1994.
DOI: 10.1016/0146-6410(94)90045-0.

[72] R. Kuchimanchi, "Solution to the Strong CP Problem: Supersymmetry with Parity", *Physical Review Letters*, vol. 76, pp. 3486–3489, 19 May 1996.
DOI: 10.1103/PhysRevLett.76.3486.

[73] "SUSY October 2019 Summary Plot Update", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2019-044, Oct. 2019.
[Online]. Available: http://cds.cern.ch/record/2697155.

[74] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Heidelberg: Springer-Verlag, 2006, ISBN: 0387310738.

[75] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
[Online]. Available: http://www.deeplearningbook.org.

[76] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. USA: Cambridge University Press, 2014, ISBN: 1107057132.

[77] M. A. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2015.
[Online]. Available: http://neuralnetworksanddeeplearning.com.

[78] T. M. Mitchell, *Machine Learning*, 1st ed. USA: McGraw-Hill, 1997, ISBN: 0070428077.

[79] M. Lindgren, C. McKay, *et al.*, *Glory and Failure: The Difference Engines of Johann Müller, Charles Babbage and Georg and Edvard Scheutz*, ser. History of computing. MIT Press, 1990.
[Online]. Available: https://books.google.co.za/books?id=plgMl2yfVkwC.

[80] J. Fuegi and J. Francis, "Lovelace & Babbage and the creation of the 1843 'notes'", *IEEE Annals of the History of Computing*, vol. 25, no. 4, pp. 16–26, 2003, ISSN: 1934-1547.
DOI: 10.1109/MAHC.2003.1253887.

[81] J. Essinger, *Ada's Algorithm: How Lord Byron's Daughter Ada Lovelace Launched the Digital Age*. Melville House, 2013, ISBN: 978-1-61219-408-0.

[82] A. A. Lovelace, "Notes by A.A.L.", *Scientific Memoirs, Selections from The Transactions of Foreign Academies and Learned Societies and from Foreign Journals*, vol. 3, no. 3, p. 722, 1843.

[83] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers", *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959.
DOI: 10.1147/rd.33.0210.

[84] T. M. Mitchell, "The Discipline of Machine Learning", SCS Technical Report Collection, School of Computer Science, Carnegie Mellon University, Pittsburgh, Tech. Rep., Jun. 2006.
[Online]. Available: http://cs.cmu.edu/~tom/pubs/MachineLearning.pdf.

[85] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", in *Proceedings, Advances in Neural Information Processing Systems, Stateline, USA*, 2012, pp. 1097–1105.
[Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[86] S. Wojcicki, *Appeal of Conscience Foundation Remarks*, Speech delivered at the Appeal of Conscience Foundation Awards Dinner, Sep. 2019.
[Online]. Available: https://youtube.googleblog.com/2019/09/appealspeech.html.

[87] P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations", in *Proceedings, ACM Conference on Recommender Systems, New York, USA*, 2016, pp. 191–198.
[Online]. Available: https://research.google/pubs/pub45530/.

[88] D. F. Specht *et al.*, "A general regression neural network", *IEEE Transactions on Neural Networks*, vol. 2, no. 6, pp. 568–576, 1991.

[89] J. K. Chorowski, D. Bahdanau, *et al.*, "Attention-based models for speech recognition", in *Proceedings, Advances in Neural Information Processing Systems, Montreal, Canada*, 2015, pp. 577–585.
arXiv: 1506.07503 [cs.CL].

[90] "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups", *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov. 2012, ISSN: 1558-0792.
DOI: 10.1109/MSP.2012.2205597.

[91] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey", *ACM Computing Surveys*, vol. 41, no. 3, p. 15, Jul. 2009, ISSN: 0360-0300.
DOI: 10.1145/1541880.1541882.

[92] T. Haarnoja, S. Ha, *et al.*, "Learning to Walk via Deep Reinforcement Learning", *ArXiv Preprint*, 2018.
arXiv: 1812.11103 [cs.LG].

[93] C. Berner, G. Brockman, B. Chan, *et al.*, "Dota 2 with Large Scale Deep Reinforcement Learning", *ArXiv Preprint*, 2019.
arXiv: 1912.06680 [cs.LG].

[94] K. Hornik, "Approximation capabilities of multilayer feedforward networks", *Neural Networks*, vol. 4, no. 2, pp. 251–257, 1991, ISSN: 0893-6080.
DOI: https://doi.org/10.1016/0893-6080(91)90009-T.

[95] R. Eldan and O. Shamir, "The power of depth for feedforward neural networks", in *Proceedings, Conference on Learning Theory, New York, USA*, 2016, pp. 907–940.
arXiv: 1512.03965 [cs.LG].

[96] M. Telgarsky, "Benefits of Depth in Neural Networks", *Journal of Machine Learning Research*, vol. 49, pp. 1–23, 2016.
arXiv: 1602.04485 [cs.LG].

[97] K. Murphy, *Machine Learning: A Probabilistic Perspective*, ser. Adaptive Computation and Machine Learning series. MIT Press, 2012, ISBN: 9780262018029.
[Online]. Available: https://books.google.co.za/books?id=NZP6AQAAQBAJ.

[98] Y. LeCun, *Who is afraid of non-convex loss functions?*, Lecture at NIPS Workshop on Efficient Machine Learning, Vancouver, Canada, 2007.
[Online]. Available: https://cs.nyu.edu/~yann/talks/lecun-20071207-nonconvex.pdf.

[99] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by backpropagating errors", *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
DOI: 10.1038/323533a0.

[100] K. Cho, B. Van Merriënboer, C. Gulcehre, *et al.*, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation", in *Proceedings, Conference on Empirical Methods in Natural Language Processing, Doha, Qatar*, Oct. 2014, pp. 1724–1734.
DOI: 10.3115/v1/D14-1179.

[101] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout", in *Proceedings, International Conference on Acoustics, Speech, and Signal Processing, Vancouver, Canada*, IEEE, 2013, pp. 8609–8613.
DOI: 10.1109/ICASSP.2013.6639346.

[102] D. Sussillo and L. Abbott, "Random walk initialization for training very deep feedforward networks", *arXiv preprint arXiv:1412.6558*, 2014.

[103] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks", in *Proceedings, International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, USA*, vol. 15, Apr. 2011, pp. 315–323.
[Online]. Available: http://proceedings.mlr.press/v15/glorot11a.html.

[104] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines", in *Proceedings, International Conference on Machine Learning, Hafia, Isreal*, Jun. 2010, pp. 807–814.
[Online]. Available: https://www.cs.toronto.edu/~hinton/absps/reluICML.pdf.

[105] L. Lu, Y. Shin, *et al.*, "Dying ReLU and Initialization: Theory and Numerical Examples", *ArXiv Preprint*, 2019.
arXiv: 1903.06733 [stat.ML].

[106] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models", in *Proceedings, International Conference on Machine Learning, Atlanta, USA*, vol. 30, 2013, p. 3.
[Online]. Available: https://awnihannun.com/papers/relu_hybrid_icml2013_final.pdf.

[107] B. Xu, N. Wang, *et al.*, "Empirical evaluation of rectified activations in convolutional network", *ArXiv Preprint*, 2015.
arXiv: 1505.00853 [cs.LG].

[108] K. He, X. Zhang, *et al.*, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification", in *Proceedings, International Conference on Computer Vision, Santiago, Chile*, 2015, pp. 1026–1034.
arXiv: 1502.01852 [cs.CV].

[109] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)", in *Proceedings, International Conference on Learning Representations, San Juan, Puerto Rico*, Feb. 2016.
arXiv: 1511.07289 [cs.LG].

[110] D. Pedamonti, "Comparison of non-linear activation functions for deep neural networks on MNIST classification task", *ArXiv Preprint*, 2018.
arXiv: 1804.02763 [cs.LG].

[111] G. Klambauer, T. Unterthiner, *et al.*, "Self-normalizing neural networks", in *Proceedings, Advances in Neural Information Processing Systems, Long Beach, USA*, 2017, pp. 971–980.
arXiv: 1706.02515 [cs.LG].

[112] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions", *ArXiv Preprint*, 2017.
arXiv: 1710.05941 [cs.NE].

[113] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning", *Neural Networks*, vol. 107, pp. 3–11, Jan. 2018.
DOI: 10.1016/j.neunet.2017.12.012.

[114] S. Ruder, "An overview of gradient descent optimization algorithms", *ArXiv Preprint*, 2016.
arXiv: 1609.04747 [cs.LG].

[115] Y. Dauphin, R. Pascanu, *et al.*, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization", in *Proceedings, Advances in Neural Information Processing Systems, Montreal, Canada*, vol. 2, 2014, pp. 2933–2941.
arXiv: 1406.2572 [cs.LG].

[116] R. S. Sutton, "Two Problems with Backpropagation and Other Steepest-Descent Learning Procedures for Networks", in *Proceedings, Annual Conference of the Cognitive Science Society, Amherst, USA*, Hillsdale, NJ: Erlbaum, 1986.
[Online]. Available: http://incompleteideas.net/papers/sutton-86.pdf.

[117] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods", *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964, ISSN: 0041-5553.
DOI: https://doi.org/10.1016/0041-5553(64)90137-5.

[118] N. Qian, "On the momentum term in gradient descent learning algorithms", *Neural Networks*, vol. 12, no. 1, pp. 145–151, 1999, ISSN: 0893-6080.
DOI: https://doi.org/10.1016/S0893-6080(98)00116-6.

[119] I. Sutskever, J. Martens, *et al.*, "On the importance of initialization and momentum in deep learning", in *Proceedings, International Conference on Machine Learning, Atlanta, USA*, vol. 28, Jun. 2013, pp. 1139–1147.
[Online]. Available: http://proceedings.mlr.press/v28/sutskever13.html.

[120] Y. E. Nesterov, "A method for solving the convex programming problem with convergence rate $O(1/k^2)$", *Soviet Mathematics Doklady*, vol. 27, no. 2, pp. 372–376, 1983.
[Online]. Available: http://mpawankumar.info/teaching/cdt-big-data/nesterov83.pdf.

[121] Y. Bengio, N. Boulanger-Lewandowski, and R. Pascanu, "Advances in Optimizing Recurrent Networks", in *Proceedings, International Conference on Acoustics, Speech, and Signal Processing, Kyoto, Japan*, May 2013, pp. 8624–8628.
DOI: 10.1109/ICASSP.2013.6639349.

[122] I. Sutskever, "Training recurrent neural networks", PhD thesis, University of Toronto, Toronto, Canada, 2013.
[Online]. Available: https://www.cs.utoronto.ca/~ilya/pubs/ilya_sutskever_phd_thesis.pdf.

[123] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization", *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, Jul. 2011.
[Online]. Available: http://www.jmlr.org/papers/volume12/duchi11a/duchi11a.pdf.

[124] J. Dean, G. Corrado, *et al.*, "Large scale distributed deep networks", in *Proceedings, Advances in Neural Information Processing Systems, Stateline, USA*, 2012, pp. 1223–1231.
[Online]. Available: http://papers.nips.cc/paper/4687-large-scale-distributed-deep-networks.pdf.

[125] G. Hinton, N. Srivastava, and K. Swersky, "Neural Networks for Machine Learning: Lecture 6a Overview of mini-batch gradient descent", *Coursera Lecture Slides*, 2012.
[Online]. Available: https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.

[126] M. D. Zeiler, "ADADELTA: An Adaptive Learning Rate Method", *ArXiv Preprint*, 2012.
arXiv: 1212.5701 [cs.LG].

[127] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", in *Proceedings, International Conference on Learning Representations, San Diego, USA*, 2015.
arXiv: 1412.6980 [cs.LG].

[128] M. Heusel, H. Ramsauer, *et al.*, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium", in *Proceedings, Advances in Neural Information Processing Systems, Long Beach, USA*, Dec. 2017, pp. 971–980.
arXiv: 1706.08500 [cs.LG].

[129] N. Srivastava, G. Hinton, *et al.*, "Dropout: a simple way to prevent neural networks from overfitting", *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958,
[Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html.

[130] Y. Gal, "Uncertainty in Deep Learning", PhD thesis, University of Cambridge, Cambridge, United kingdom, 2016.
[Online]. Available: http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf.

[131] R. Pascanu and Y. Bengio, "Revisiting natural gradient for deep networks", in *International Conference on Learning Representations, Scottsdale, USA*, 2013.
arXiv: 1301.3584 [cs.LG].

[132] A. Paszke, S. Gross, *et al.*, "Automatic differentiation in PyTorch", 2017, Software available from pytorch.org.
[Online]. Available: pytorch.org.

[133] M. Abadi, A. Agarwal, *et al.*, *TensorFlow: Large-scale machine learning on heterogeneous systems*, Software available from tensorflow.org, 2015.
[Online]. Available: http://tensorflow.org/.

[134] J. Ling, R. Jones, and J. Templeton, "Machine learning strategies for systems with invariance properties", *Journal of Computational Physics*, vol. 318, pp. 22–35, 2016, ISSN: 0021-9991.
DOI: https://doi.org/10.1016/j.jcp.2016.05.003.

[135] F. Schilling, "The effect of batch normalization on deep convolutional neural networks", Dissertation, KTH Royal Institute of Technology, 2016.

[136] S. Santurkar, D. Tsipras, *et al.*, "How does batch normalization help optimization?", in *Proceedings, Advances in Neural Information Processing Systems, Montreal, Canada*, 2018, pp. 2483–2493.
arXiv: 1805.11604 [stat.ML].

[137] P. Luo, X. Wang, *et al.*, "Towards understanding regularization in batch normalization", 2019.
[Online]. Available: https://openreview.net/pdf?id=HJlLKjR9FQ.

[138] N. Bjorck, C. P. Gomes, *et al.*, "Understanding batch normalization", in *Proceedings, Advances in Neural Information Processing Systems, Montreal, Canada*, 2018, pp. 7694–7705.
arXiv: 1806.02375 [cs.LG].

[139] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", in *International Conference on International Conference on Machine Learning, Lille, France*, 2015, pp. 448–456.
arXiv: 1502.03167 [cs.LG].

[140] A. C. Wilson, R. Roelofs, *et al.*, "The Marginal Value of Adaptive Gradient Methods in Machine Learning", in *Advances in Neural Information Processing Systems, Long Beach, USA*, I. Guyon, U. V. Luxburg, *et al.*, Eds., Curran Associates, Inc., 2017, pp. 4148–4158.
arXiv: 1705.08292 [stat.ML].

[141] "About CERN", Jan. 2012.
[Online]. Available: http://cds.cern.ch/record/1997225 (visited 29/1/2019).

[142] G. Arnison *et al.*, "Experimental observation of isolated large transverse energy electrons with associated missing energy at $s = 540$ GeV", *Physics Letters*, vol. B122, no. 1, pp. 103–116, 1983, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/0370-2693(83)91177-2.

[143] P. Bagnaia *et al.*, "Evidence for $Z_0 \to e^+e^-$ at the CERN anti-pp collider", *Physics Letters*, vol. B129, pp. 130–140, 1983.
DOI: 10.1016/0370-2693(83)90744-X.

[144] "CERN Annual report 2017", CERN, Geneva, Tech. Rep., 2018.
[Online]. Available: https://cds.cern.ch/record/2624296.

[145] *Worldwide LHC Computing Grid: About page*.
[Online]. Available: http://wlcg-public.web.cern.ch/about (visited 30/1/2019).

[146] M. Giampietro, "The World Wide Web's 25th anniversary", *CERN Courier*, vol. 54, no. 4, pp. 27–30, May 2014.
[Online]. Available: http://cds.cern.ch/record/2064554.

[147] E. Mobs, "The CERN accelerator complex", Aug. 2018, General Photo.
[Online]. Available: https://cds.cern.ch/record/2636343.

[148] O. S. Bruning, P. Collier, *et al.*, "LHC Design Report Vol.1: The LHC Main Ring", CERN Yellow Reports: Monographs, vol. 1, 2004.
DOI: 10.5170/CERN-2004-003-V-1.

[149] ALICE Collaboration, *Measuring the highest temperature on Earth*, Jul. 2014.
[Online]. Available: http://alice.web.cern.ch/content/measuring-highest-temperature-earth (visited 11/1/2019).

[150] ALICE Collaboration, "Direct photon production in Pb–Pb collisions at $\sqrt{NN} = 2.76$ TeV", *Physics Letters*, vol. B754, pp. 235–248, 2016, ISSN: 0370-2693.
DOI: https://doi.org/10.1016/j.physletb.2016.01.020.

[151] Guinness World Records News, *Higgs boson discovery: The top ten Large Hadron Collider world records*, Jul. 2012.
[Online]. Available: http://www.guinnessworldrecords.com/news/2012/7/higgs-boson-discovery-the-top-ten-large-hadron-collider-world-records-43350 (visited 11/1/2019).

[152] L. Evans and P. Bryant, "LHC Machine", *Journal of Instrumentation*, vol. 3, no. 08, S08001, 2008.
DOI: 10.1088/1748-0221/3/08/S08001.

[153] ATLAS Collaboration, "The ATLAS Experiment at the CERN Large Hadron Collider", *Journal of Instrumentation*, vol. 3, no. 08, S08003, 2008.
DOI: 10.1088/1748-0221/3/08/S08003.

[154] CMS Collaboration, "The CMS experiment at the CERN LHC", *Journal of Instrumentation*, vol. 3, no. 08, S08004, 2008.
DOI: 10.1088/1748-0221/3/08/S08004.

[155] ALICE Collaboration, "The ALICE experiment at the CERN LHC", *Journal of Instrumentation*, vol. 3, no. 08, S08002, 2008.
DOI: 10.1088/1748-0221/3/08/S08002.

[156] The LHCb Collaboration, "The LHCb Detector at the LHC", *Journal of Instrumentation*, vol. 3, no. 08, S08005, 2008.
DOI: 10.1088/1748-0221/3/08/S08005.

[157] X. Vidal and R. Manzano, *LHC layout: Taking a closer look at LHC*, General Photo.
[Online]. Available: https://www.lhc-closer.es/taking_a_closer_look_at_lhc/0.lhc_layout (visited 11/2/2019).

[158] The LHCf Collaboration, "The LHCf detector at the CERN Large Hadron Collider", *Journal of Instrumentation*, vol. 3, no. 08, S08006, 2008.
DOI: 10.1088/1748-0221/3/08/S08006.

[159] The TOTEM Collaboration, "The TOTEM Experiment at the CERN Large Hadron Collider", *Journal of Instrumentation*, vol. 3, no. 08, S08007, 2008.
DOI: 10.1088/1748-0221/3/08/S08007.

[160] J. Pinfold, R. Soluk, *et al.*, "Technical Design Report of the MoEDAL Experiment", Tech. Rep., Jun. 2009.
[Online]. Available: http://cds.cern.ch/record/1181486.

[161] W. Herr and B. Muratori, "Concept of luminosity", in *Proceedings, Intermediate accelerator physics, CERN Accelerator School, Zeuthen, Germany*, Sep. 2003, pp. 361–377.
[Online]. Available: http://doc.cern.ch/yellowrep/2006/2006-002/p361.pdf.

[162] G. Barr, R. Devenish, *et al.*, *Particle Physics in the LHC Era*. Oxford: Oxford University Press, 2016, vol. 24, pp. 63–64, ISBN: 3257227892.
DOI: 10.1093/acprof:oso/9780198748557.001.0001.

[163] M. Thomson, *Modern Particle Physics*. Cambridge University Press, 2013, pp. 25–27, ISBN: 9781107034266.

[164] "LHC smashes luminosity record", *Physics World*, vol. 29, no. 12, p. 13, Dec. 2016.
DOI: 10.1088/2058-7058/29/12/28.

[165] J. Wenninger and M. Hostettler, *LHC Report: colliding at an angle*, Sep. 2017.
[Online]. Available: https://home.cern/news/news/accelerators/lhc-report-colliding-angle (visited 5/2/2019).

[166] J. Wenninger and M. Hostettler, *Record luminosity: well done LHC*, Nov. 2017.
[Online]. Available: https://home.cern/news/news/accelerators/record-luminosity-well-done-lhc (visited 5/2/2019).

[167] R. Steerenberg, *LHC Report: Another run is over and LS2 has just begun...* Dec. 2018.
[Online]. Available: https://home.cern/news/news/accelerators/lhc-report-another-run-over-and-ls2-has-just-begun (visited 5/2/2019).

[168] ATLAS Collaboration, *ATLAS inner detector: Technical Design Report, 1*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997.
[Online]. Available: https://cds.cern.ch/record/331063.

[169] ATLAS Collaboration, *ATLAS inner detector: Technical Design Report, 2*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997.
[Online]. Available: https://cds.cern.ch/record/331064.

[170] ATLAS Collaboration, *ATLAS calorimeter performance: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1996.
[Online]. Available: https://cds.cern.ch/record/331059.

[171] ATLAS Collaboration, *ATLAS muon spectrometer: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997.
[Online]. Available: https://cds.cern.ch/record/331068.

[172] ATLAS Collaboration, *ATLAS central solenoid: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997, Electronic version not available.
[Online]. Available: https://cds.cern.ch/record/331067.

[173] ATLAS Collaboration, *ATLAS barrel toroid: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997, Electronic version not available.
[Online]. Available: https://cds.cern.ch/record/331065.

[174] ATLAS Collaboration, *ATLAS end-cap toroids: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1997, Electronic version not available.
[Online]. Available: https://cds.cern.ch/record/331066.

[175] ATLAS Collaboration, *ATLAS pixel detector: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1998.
[Online]. Available: https://cds.cern.ch/record/381263.

[176] ATLAS Collaboration, "ATLAS pixel detector electronics and sensors", *Journal of Instrumentation*, vol. 3, no. 07, P07007, Jun. 2008.
DOI: 10.1088/1748-0221/3/07/p07007.

[177] A. Abdesselam *et al.*, "The barrel modules of the ATLAS semiconductor tracker", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 568, no. 2, pp. 642–671, 2006, ISSN: 0168-9002.
DOI: https://doi.org/10.1016/j.nima.2006.08.036.

[178] A. Abdesselam *et al.*, "The ATLAS semiconductor tracker end-cap module", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 575, no. 3, pp. 353–389, 2007, ISSN: 0168-9002.
DOI: https://doi.org/10.1016/j.nima.2007.02.019.

[179] A. Ahmad *et al.*, "The silicon microstrip sensors of the atlas semiconductor tracker", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 578, no. 1, pp. 98–118, 2007, ISSN: 0168-9002.
DOI: https://doi.org/10.1016/j.nima.2007.04.157.

[180] The ATLAS TRT collaboration, "The ATLAS TRT Barrel Detector", *Journal of Instrumentation*, vol. 3, no. 02, P02014–P02014, Feb. 2008.
DOI: 10.1088/1748-0221/3/02/p02014.

[181] The ATLAS TRT collaboration, "The ATLAS TRT end-cap detectors", *Journal of Instrumentation*, vol. 3, no. 10, P10003–P10003, Oct. 2008.
DOI: 10.1088/1748-0221/3/10/p10003.

[182] The ATLAS TRT collaboration, "The ATLAS Transition Radiation Tracker (TRT) proportional drift tube: design and performance", *Journal of Instrumentation*, vol. 3, no. 02, P02013–P02013, Feb. 2008.
DOI: 10.1088/1748-0221/3/02/p02013.

[183] F. Ahmadov *et al.*, "The ATLAS Inner Detector commissioning and calibration", *The European Physical Journal*, vol. C70, pp. 787–821, Dec. 2010.
DOI: 10.1140/epjc/s10052-010-1366-7.

[184] M. Capeans, G. Darbo, *et al.*, "ATLAS Insertable B-Layer Technical Design Report", Tech. Rep. CERN-LHCC-2010-013. ATLAS-TDR-19, Sep. 2010.
[Online]. Available: https://cds.cern.ch/record/1291633.

[185] K. Potamianos, "The upgraded Pixel detector and the commissioning of the Inner Detector tracking of the ATLAS experiment for Run-2 at the Large Hadron Collider", in *Proceedings, European Physical Society Conference on High Energy Physics, Vienna, Austria*, vol. EPS-HEP2015, Jul. 2015, p. 261.
arXiv: 1608.07850 [physics.ins-det].

[186] V. A. Mitsou, "The ATLAS transition radiation tracker", in *Astroparticle, Particle And Space Physics, Detectors And Medical Physics Applications*, World Scientific, 2004, pp. 497–501.

[187] M. Kayl, "Tracking Performance of the ATLAS Inner Detector and Observation of Known Hadrons", in *Proceedings, Hadron collider physics, Toronto, Canada*, Aug. 2010.
arXiv: 1010.1091 [physics.ins-det].

[188] F. Bauer *et al.*, "Construction and test of MDT chambers for the ATLAS muon spectrometer", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 461, no. 1, pp. 17–20, 2001, 8th Pisa Meeting on Advanced Detectors, ISSN: 0168-9002.
DOI: https://doi.org/10.1016/S0168-9002(00)01156-6.

[189] T. Argyropoulos *et al.*, "Cathode Strip Chambers in ATLAS: Installation, Commissioning and in Situ Performance", *IEEE Transactions on Nuclear Science*, vol. 56, no. 3, pp. 1568–1574, Jun. 2009, ISSN: 0018-9499.
DOI: 10.1109/TNS.2009.2020861.

[190] G. Aielli *et al.*, "The RPC first level muon trigger in the barrel of the ATLAS experiment", *Nuclear Physics B - Proceedings Supplements*, vol. 158, pp. 11–15, 2006, Proceedings of the 8th International Workshop on Resistive Plate Chambers and Related Detectors, ISSN: 0920-5632.
DOI: https://doi.org/10.1016/j.nuclphysbps.2006.07.031.

[191] S. Majewski, G. Charpak, *et al.*, "A thin multiwire chamber operating in the high multiplication mode", *Nuclear Instruments and Methods in Physics Research*, vol. 217, no. 1, pp. 265–271, 1983, ISSN: 0167-5087.
DOI: https://doi.org/10.1016/0167-5087(83)90146-1.

[192] E. Diehl, "Calibration and Performance of the ATLAS Muon Spectrometer", in *Proceedings, Meeting of the Division of the American Physical Society, Providence, USA*, Aug. 2011.
arXiv: 1109.6933 [physics.ins-det].

[193] ATLAS Collaboration, "Standalone Vertex Finding in the ATLAS Muon Spectrometer", *JINST*, vol. 9, no. CERN-PH-EP-2013-185. CERN-PH-EP-2013-185, P02001. 22 p, Nov. 2013.
[Online]. Available: http://cds.cern.ch/record/1631701.

[194] A. R. Martínez, "The Run-2 ATLAS Trigger System", in *Proceedings, International Workshop on Advanced Computing and Analysis Techniques in Physics Research, Valparaiso, Chile*, vol. 762, Jan. 2016, p. 012 003.
DOI: 10.1088/1742-6596/762/1/012003.

[195] ATLAS Collaboration, *ATLAS level-1 trigger: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 1998.
[Online]. Available: http://cds.cern.ch/record/381429.

[196] P. Jenni, M. Nessi, *et al.*, *ATLAS high-level trigger, data-acquisition and controls: Technical Design Report*, ser. Technical Design Report ATLAS. Geneva: CERN, 2003.
[Online]. Available: http://cds.cern.ch/record/616089.

[197] Z. M. and, "Simulation of Pile-up in the ATLAS Experiment", *Journal of Physics: Conference Series*, vol. 513, no. 2, p. 022 024, Jun. 2014.
DOI: 10.1088/1742-6596/513/2/022024.

[198] ATLAS Collaboration, "Luminosity determination in pp collisions at $\sqrt{s} = 13$ TeV using the ATLAS detector at the LHC", CERN, Geneva, Tech. Rep. ATLAS-CONF-2019-021, Jun. 2019.
[Online]. Available: http://cds.cern.ch/record/2677054.

[199] ATLAS Collaboration, "Commissioning of the ATLAS Muon Spectrometer with cosmic rays", *The European Physical Journal*, vol. C70, no. 3, pp. 875–916, Dec. 2010, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-010-1415-2.

[200] T. Cornelissen, M. Elsing, *et al.*, "Concepts, Design and Implementation of the ATLAS New Tracking (NEWT)", CERN, Geneva, Tech. Rep. ATL-SOFT-PUB-2007-007. ATL-COM-SOFT-2007-002, Mar. 2007.
[Online]. Available: https://cds.cern.ch/record/1020106.

[201] ATLAS Collaboration, "Reconstruction of primary vertices at the ATLAS experiment in Run 1 proton–proton collisions at the LHC", *The European Physical Journal*, vol. C77, p. 332, May 2017.
DOI: 10.1140/epjc/s10052-017-4887-5.

[202] ATLAS collaboration, "Topological cell clustering in the ATLAS calorimeters and its performance in LHC Run 1", *The European Physical Journal*, vol. C77, no. CERN-PH-EP-2015-304, 490. 87 p, Mar. 2016.
DOI: 10.1140/epjc/s10052-017-5004-5.

[203] M. Cacciari, G. P. Salam, and G. Soyez, "The anti-k_t jet clustering algorithm", *Journal of High Energy Physics*, vol. 04, p. 063, 2008.
DOI: 10.1088/1126-6708/2008/04/063.

[204] ATLAS Collaboration, "Commissioning of the ATLAS high-performance b-tagging algorithms in the 7 TeV collision data", CERN, Geneva, Tech. Rep. ATLAS-CONF-2011-102, Jul. 2011.
[Online]. Available: https://cds.cern.ch/record/1369219.

[205] ATLAS Collaboration, "Electron reconstruction and identification in the ATLAS experiment using the 2015 and 2016 LHC proton-proton collision data at $\sqrt{s}$ = 13 TeV", *The European Physical Journal*, vol. C79, no. 8, p. 639, 2019.
DOI: 10.1140/epjc/s10052-019-7140-6.

[206] ATLAS Collaboration, "Muon reconstruction performance of the ATLAS detector in proton–proton collision data at $\sqrt{s}$=13 TeV", *The European Physical Journal*, vol. C76, no. 5, p. 292, 2016.
DOI: 10.1140/epjc/s10052-016-4120-y.

[207] ATLAS Collaboration, "Muon reconstruction performance in early $\sqrt{s} = 13$ TeV data", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2015-037, Aug. 2015.
[Online]. Available: http://cds.cern.ch/record/2047831.

[208] Z. van Kesteren, "Identification of muons in ATLAS", Presented on 12 Mar 2010, PhD thesis, University of Amsterdam, 2010.
[Online]. Available: https://cds.cern.ch/record/1255858.

[209] O. Kortner, "Searches for the Supersymmetric Partner of the Top Quark, Dark Matter and Dark Energy at the ATLAS Experiment", PhD thesis, ATLAS, Canada, 2019.
DOI: 10.1007/978-3-030-25988-4.

[210] ATLAS Collaboration, "Electron efficiency measurements with the ATLAS detector using the 2015 LHC proton-proton collision data", CERN, Geneva, Tech. Rep. ATLAS-CONF-2016-024, Jun. 2016.
[Online]. Available: http://cds.cern.ch/record/2157687.

[211] ATLAS Collaboration, "Measurements of the photon identification efficiency with the ATLAS detector using 4.9 fb^{-1} of pp collision data collected in 2011", CERN, Geneva, Tech. Rep. ATLAS-CONF-2012-123, Aug. 2012.
[Online]. Available: https://cds.cern.ch/record/1473426.

[212] ATLAS Collaboration, "Measurement of the photon identification efficiencies with the ATLAS detector using LHC Run 2 data collected in 2015 and 2016", *The European Physical Journal*, vol. C79, no. 3, p. 205, 2019.
DOI: 10.1140/epjc/s10052-019-6650-6.

[213] G. P. Salam, "Towards Jetography", *The European Physical Journal*, vol. C67, pp. 637–686, 2010.
DOI: 10.1140/epjc/s10052-010-1314-6.

[214] ATLAS Collaboration, "Jet Calibration and Systematic Uncertainties for Jets Reconstructed in the ATLAS Detector at $\sqrt{s} = 13$ TeV", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2015-015, Jul. 2015.
[Online]. Available: https://cds.cern.ch/record/2037613.

[215] ATLAS Collaboration, "Data-driven determination of the energy scale and resolution of jets reconstructed in the ATLAS calorimeters using dijet and multijet events at $\sqrt{s} = 8$ *TeV*", CERN, Geneva, Tech. Rep. ATLAS-CONF-2015-017, Apr. 2015.
[Online]. Available: https://cds.cern.ch/record/2008678.

[216] ATLAS Collaboration, "Determination of the jet energy scale and resolution at ATLAS using Z/γ–jet events in data at $\sqrt{s} = 8$ TeV", CERN, Geneva, Tech. Rep. ATLAS-CONF-2015-057, Oct. 2015.
[Online]. Available: https://cds.cern.ch/record/2059846.

[217] ATLAS Collaboration, "Tagging and suppression of pileup jets with the ATLAS detector", CERN, Geneva, Tech. Rep. ATLAS-CONF-2014-018, May 2014.
[Online]. Available: https://cds.cern.ch/record/1700870.

[218] ATLAS Collaboration, "Forward Jet Vertex Tagging: A new technique for the identification and rejection of forward pileup jets", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2015-034, Aug. 2015.
[Online]. Available: https://cds.cern.ch/record/2042098.

[219] G. Piacquadio and C. Weiser, "A new inclusive secondary vertex algorithm for b-jet tagging in ATLAS", *Journal of Physics: Conference Series*, vol. 119, no. 3, p. 032 032, Jun. 2008.
DOI: 10.1088/1742-6596/119/3/032032.

[220] ATLAS Collaboration, "Optimisation of the ATLAS b-tagging performance for the 2016 LHC Run", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2016-012, Jun. 2016.
[Online]. Available: https://cds.cern.ch/record/2160731.

[221] ATLAS Collaboration, "Optimisation of the ATLAS b-tagging performance for the 2016 LHC Run", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2016-012, Jun. 2016.
[Online]. Available: https://cds.cern.ch/record/2160731.

[222] ATLAS Collaboration, "Search for electroweak production of supersymmetric particles in final states with two or three leptons at $\sqrt{s} = 13$ TeV with the ATLAS detector", *The European Physical Journal*, vol. C78, no. 12, p. 995, Dec. 2018, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-018-6423-7.

[223] S. Farrell, "Overlap Removal Tools", ATLAS Joint Flavour Tagging and $H \to bb$ Workshop 2017, 2017.
[Online]. Available: http://sbhep.physics.sunysb.edu/HEP/ATLASbbWorkshop2017/index.html.

[224] ATLAS Collaboration, "Object-based missing transverse momentum significance in the ATLAS detector", CERN, Geneva, Tech. Rep. ATLAS-CONF-2018-038, Jul. 2018.
[Online]. Available: https://cds.cern.ch/record/2630948.

[225] ATLAS Collaboration, "Data-driven determination of the energy scale and resolution of jets reconstructed in the ATLAS calorimeters using dijet and multijet events at $\sqrt{s} = 8$ TeV", CERN, Geneva, Tech. Rep. ATLAS-CONF-2015-017, Apr. 2015.
[Online]. Available: http://cds.cern.ch/record/2008678.

[226] ATLAS Collaboration, "Jet energy measurement with the ATLAS detector in proton-proton collisions at $\sqrt{s} = 7$ TeV", *The European Physical Journal*, vol. C73, no. 3, Mar. 2013, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-013-2304-2.

[227] ATLAS Collaboration, "Electron and photon energy calibration with the ATLAS detector using LHC Run 1 data", *The European Physical Journal*, vol. C74, no. 10, p. 3071, 2014.
DOI: 10.1140/epjc/s10052-014-3071-4.

[228] ATLAS collaboration, "Jet energy scale measurements and their systematic uncertainties in proton-proton collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector", *Physical Review*, vol. D96, no. CERN-EP-2017-038. 7, p. 36, Mar. 2017.
DOI: 10.1103/PhysRevD.96.072002.

[229] ATLAS Collaboration, "Performance of missing transverse momentum reconstruction in proton-proton collisions at $\sqrt{s} = 7$ TeV with ATLAS", *The European Physical Journal*, vol. C72, no. 1, Jan. 2012, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-011-1844-6.

[230] ATLAS Collaboration, "Performance of the ATLAS Inner Detector Track and Vertex Reconstruction in the High Pile-Up LHC Environment", CERN, Geneva, Tech. Rep. ATLAS-CONF-2012-042, Mar. 2012.
[Online]. Available: https://cds.cern.ch/record/1435196.

[231] ATLAS Collaboration, "Evidence for the Higgs boson Yukawa coupling to tau leptons with the ATLAS detector", *Journal of High Energy Physics*, vol. 2015, no. 4, Apr. 2015, ISSN: 1029-8479.
DOI: 10.1007/jhep04(2015)117.

[232] ATLAS Collaboration, "The ATLAS Simulation Infrastructure", *The European Physical Journal*, vol. C70, no. 3, pp. 823–874, Sep. 2010, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-010-1429-9.

[233] S. Frixione, P. Nason, and C. Oleari, "Matching NLO QCD computations with Parton Shower simulations: the POWHEG method", *Journal of High Energy Physics*, vol. 11, p. 070, 2007.
DOI: 10.1088/1126-6708/2007/11/070.

[234] T. Sjostrand, S. Mrenna, and P. Z. Skands, "A Brief Introduction to PYTHIA 8.1", *Computer Physics Communications*, vol. 178, pp. 852–867, 2008.
DOI: 10.1016/j.cpc.2008.01.036.

[235] S. Schumann and F. Krauss, "A parton shower algorithm based on Catani-Seymour dipole factorisation", *Journal of High Energy Physics*, vol. 2008, no. 03, pp. 038–038, Mar. 2008, ISSN: 1029-8479.
DOI: 10.1088/1126-6708/2008/03/038.

[236] S. Höche, F. Krauss, *et al.*, "QCD matrix elements + parton showers. The NLO case", *Journal of High Energy Physics*, vol. 2013, no. 4, Apr. 2013, ISSN: 1029-8479.
DOI: 10.1007/jhep04(2013)027.

[237] D. J. Lange, "The EvtGen particle decay simulation package", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 462, no. 1, pp. 152–155, 2001, BEAUTY2000, Proceedings of the 7th Int. Conf. on B-Physics at Hadron Machines, ISSN: 0168-9002.
DOI: https://doi.org/10.1016/S0168-9002(01)00089-4.

[238] ATLAS Collaboration, "Measurement of the Z/γ^* boson transverse momentum distribution in pp collisions at $\sqrt{s} = 7$ TeV with the ATLAS detector", *Journal of High Energy Physics*, vol. 09, p. 145, 2014.
DOI: 10.1007/JHEP09(2014)145.

[239] J. Pumplin, D. R. Stump, *et al.*, "New Generation of Parton Distributions with Uncertainties from Global QCD Analysis", *Journal of High Energy Physics*, vol. 2002, no. 07, pp. 012–012, Jun. 2002, ISSN: 1029-8479.
DOI: 10.1088/1126-6708/2002/07/012.

[240] ATLAS Collaboration, "ATLAS Run 1 Pythia8 tunes", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2014-021, Nov. 2014.
[Online]. Available: https://cds.cern.ch/record/1966419.

[241] R. D. Ball, V. Bertone, *et al.*, "Parton distributions with LHC data", *Nuclear Physics B*, vol. 867, no. 2, pp. 244–289, 2013, ISSN: 0550-3213.
DOI: https://doi.org/10.1016/j.nuclphysb.2012.10.003.

[242] M. Czakon and A. Mitov, "Top++: A Program for the Calculation of the Top-Pair Cross-Section at Hadron Colliders", *Computer Physics Communications*, vol. 185, p. 2930, 2014.
DOI: 10.1016/j.cpc.2014.06.021.

[243] ATLAS Collaboration, "Simulation of top quark production for the ATLAS experiment at sqrt(s) = 13 TeV", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2016-004, Jan. 2016.
[Online]. Available: http://cds.cern.ch/record/2120417.

[244] R. D. Ball, V. Bertone, *et al.*, "Parton distributions for the LHC run II", *Journal of High Energy Physics*, vol. 2015, no. 4, Apr. 2015, ISSN: 1029-8479.
DOI: 10.1007/jhep04(2015)040.

[245] J. Alwall, R. Frederix, *et al.*, "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations", *Journal of High Energy Physics*, vol. 2014, no. 7, Jun. 2014, ISSN: 1029-8479.
DOI: 10.1007/jhep07(2014)079.

[246] W. Beenakker, R. Höpker, *et al.*, "Squark and gluino production at hadron colliders", *Nuclear Physics*, vol. B492, no. 1-2, pp. 51–103, May 1997, ISSN: 0550-3213.
DOI: 10.1016/s0550-3213(97)80027-2.

[247] ATLAS Collaboration, "A study of the Pythia 8 description of ATLAS minimum bias measurements with the Donnachie-Landshoff diffractive model", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2016-017, Aug. 2016.
[Online]. Available: https://cds.cern.ch/record/2206965.

[248] ATLAS Collaboration, "Identification of Jets Containing *b*-Hadrons with Recurrent Neural Networks at the ATLAS Experiment", CERN, Geneva, Tech. Rep. ATL-PHYS-PUB-2017-003, Mar. 2017.
[Online]. Available: http://cds.cern.ch/record/2255226.

[249] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization", *ArXiv Preprint*, 2016.
arXiv: 1607.06450 [stat.ML].

[250] L. Wan, M. Zeiler, *et al.*, "Regularization of neural networks using dropconnect", in *Proceedings, International Conference on Machine Learning, Atlanta, USA*, vol. 28, Jun. 2013, pp. 1058–1066.
[Online]. Available: http://proceedings.mlr.press/v28/wan13.html.

[251] C. Lester and D. Summers, "Measuring masses of semi-invisibly decaying particle pairs produced at hadron colliders", *Physics Letters*, vol. B463, no. 1, pp. 99–103, Sep. 1999, ISSN: 0370-2693.
DOI: 10.1016/s0370-2693(99)00945-4.

[252] A. Barr, C. Lester, and P. Stephens, "A variable for measuring masses at hadron colliders when missing energy is expected; Mt2: the truth behind the glamour", *Journal of Physics G: Nuclear and Particle Physics*, vol. 29, no. 10, pp. 2343–2363, Sep. 2003, ISSN: 1361-6471.
DOI: 10.1088/0954-3899/29/10/304.

[253] C. G. Lester and B. Nachman, "Bisection-based asymmetric MT2 computation: a higher precision calculator than existing symmetric methods", *Journal of High Energy Physics*, vol. 2015, no. 3, Mar. 2015, ISSN: 1029-8479.
DOI: 10.1007/jhep03(2015)100.

[254] ATLAS Collaboration, "Jet reconstruction and performance using particle flow with the ATLAS Detector", *The European Physical Journal*, vol. C77, no. 7, Jul. 2017, ISSN: 1434-6052.
DOI: 10.1140/epjc/s10052-017-5031-2.

[255] W. S. Sarle, "How to measure importance of inputs?", SAS Institute Inc, Tech. Rep., 1997. [Online]. Available: ftp://ftp.sas.com/pub/neural/importance.html.

[256] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational Inference: A Review for Statisticians", *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 859–877, Feb. 2017, ISSN: 1537-274X.
DOI: 10.1080/01621459.2017.1285773.